



TUGAS AKHIR -SM141501

DETEKSI DAN PERHITUNGAN JUMLAH KENDARAAN BERGERAK PADA WAKTU MALAM HARI BERBASIS PENGOLAHAN CITRA DIGITAL

GERY DIAS CLAUDIO
NRP 1213 100 091

Dosen Pembimbing
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

DEPARTEMEN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT -SM141501

NIGHT-TIME MOVING VEHICLE DETECTION AND COUNTING BASED DIGITAL IMAGE PROCESSING

GERY DIAS CLAUDIO
NRP 1213 100 091

Supervisor
Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

DEPARTMENT OF MATHEMATICS
Faculty of Mathematics and Natural Science
Sepuluh Nopember Institute of Technology
Surabaya 2017

LEMBAR PENGESAHAN

DETEKSI DAN PERHITUNGAN JUMLAH KENDARAAN BERGERAK PADA WAKTU MALAM HARI BERBASIS PENGOLAHAN CITRA DIGITAL

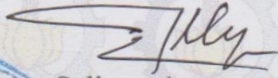
NIGHT-TIME MOVING VEHICLE DETECTION AND COUNTING BASED DIGITAL IMAGE PROCESSING

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Sains
Pada Bidang Studi Ilmu Komputer
Program Studi S-1 Departemen Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember Surabaya

Oleh :
GERY DIAS CLAUDIO
NRP. 1213 100 091

Menyetujui,
Dosen Pembimbing,



Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

Mengetahui,
Kepala Departemen Matematika
FMIPA ITS



Departemen Matematika, S.Si, MT
NRP. 1213 100 091

Surabaya, Agustus 2017

DETEKSI DAN PERHITUNGAN JUMLAH KENDARAAN BERGERAK PADA WAKTU MALAM HARI BERBASIS PENGOLAHAN CITRA DIGITAL

Nama Mahasiswa : Gery Dias Claudio
NRP : 1213 100 091
Jurusan : Matematika
Dosen Pembimbing : Dr. Dwi Ratna Sulistyaningrum, S.Si, MT

Abstrak

Pertumbuhan sektor transportasi di Indonesia sangat tinggi sehingga padatnya kendaraan yang berada di jalan raya. Dengan jumlah kendaraan yang semakin bertambah, fasilitas jalan saat ini tidak dapat mengimbangi sehingga terjadinya permasalahan-permasalahan lalu lintas. Salah satu cara untuk mengatasi permasalahan tersebut dengan cara pengawasan lalu lintas. Penelitian tentang pengawasan lalu lintas di jalan raya telah dilakukan oleh beberapa peneliti yang dilakukan pada pagi dan sore hari yang memiliki pencahayaan yang cukup. Akan tetapi malam hari merupakan kondisi dimana banyaknya terjadi permasalahan lalu lintas di jalan raya dikarenakan intensitas cahaya yang kurang. Maka dari itu dibutuhkan informasi kendaraan yang melintasi jalan raya pada waktu malam hari yang mampu mendeteksi dan menghitung jumlah kendaraan yang lewat di jalan raya. Pendeteksian dilakukan dengan menggunakan lampu kendaraan sebagai objek pendeteksian. Kemudian dengan terdeteksinya lampu kendaraan dapat dilakukannya perhitungan. Sehingga didapatkan hasil perhitungan kendaraan yang melewati jalan raya pada waktu malam hari.

Kata Kunci : *Pengolahan Citra, Deteksi Kendaraan, Kendaraan Malam Hari, Perhitungan Jumlah Kendaraan.*

NIGHT-TIME MOVING VEHICLE DETECTION AND COUNTING BASED DIGITAL IMAGE PROCESSING

Nama Mahasiswa : Gery Dias Claudio
NRP : 1213 100 091
Jurusan : Matematika
Dosen Pembimbing : Dr. Dwi Ratna Sulistyaningrum, S.Si,MT

Abstract

The transportation sector's growth in Indonesia is very high so that the density of vehicles that are in the road. With the increasing number of vehicles, road facilities currently can not be offset so that the occurrence of traffic problems. One way to overcome these problems by means of traffic control. Research on traffic control highway has done by some researchers. Research on road traffic monitoring has been conducted by several researchers conducted in the morning and noon that has sufficient lighting. But the evening is a condition in which the number of traffic problems occurred on the highway due to the intensity of the light is less. Therefore the required information of vehicles that cross the highway at night that is able to detect and calculate the number of vehicles passing on the highway. Detection is done by using vehicle lights as the object of detection. Then with the detection of vehicle lights can do the calculation. So get the result of vehicle counting passing the highway at night.

Keywords : ***Image Processing, Vehicle Detection, Night-Time Vehicle Detection, Vehicle Counting***

KATA PENGANTAR

Segala puji syukur penulis panjatkan ke hadirat Allah SWT, karena dengan ridho-Nya penulis dapat menyelesaikan Tugas Akhir yang berjudul

“DETEKSI DAN PERHITUNGAN JUMLAH KENDARAAN BERGERAK PADA WAKTU MALAM HARI BERBASIS PENGOLAHAN CITRA DIGITAL”

yang merupakan salah satu persyaratan akademis dalam menyelesaikan Program Sarjana Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember Surabaya.

Tugas Akhir ini dapat diselesaikan dengan baik berkat kerja sama, bantuan, dan dukungan dari banyak pihak. Sehubungan dengan hal tersebut, penulis ingin mengucapkan terima kasih kepada :

1. Dr. Imam Mukhlash, S.Si, MT selaku Kepala Departemen Matematika ITS.
2. Dr. Budi Setiyono, S.Si, MT selaku Dosen Wali yang telah memberikan arahan akademik selama penulis menempuh pendidikan di Departemen Matematika ITS.
3. Dr. Dwi Ratna Sulistyaningrum, S.Si, MT selaku Dosen Pembimbing yang telah memberikan bimbingan dan motivasi kepada penulis dalam mengerjakan Tugas Akhir ini sehingga dapat terselesaikan dengan baik.
4. Dr. Didik Khusnul Arif, S.Si, M.Si selaku Ketua Program Studi S1 Departemen Matematika ITS.
5. Drs. Iis Herisman, M.Si selaku Sekretaris Program Studi S1 Departemen Matematika ITS.
6. Seluruh jajaran dosen dan staff Departemen Matematika ITS.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kesempurnaan. Oleh karena itu, penulis mengharapkan kritik dan saran dari pembaca. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat bagi semua pihak yang berkepentingan.

Surabaya, Juli 2017

Penulis

Special thanks to

1. Bapak dan Ibu tercinta, Bapak Afliyus dan Ibu Renny Idrus yang selalu ikhlas memberikan dukungan, semangat, do'a dan nasehat-nasehat yang tiada henti kepada penulis yang tak pernah penulis lupakan.
2. Nenekku tercinta, yang selalu mengingatkan, memberikan nasehat, memberi semangat, memberi dukungan kepada penulis.
3. Saudara-saudara perempuan yang saya sayangi, Febrina Yolanda, Ranny Yolanda, Mealysa Fridayana dan Pretty Sakta Riny yang selalu memberi semangat dan dukungan kepada penulis sehingga Tugas Akhir ini dapat selesai dengan baik.
4. Ivan dan Wawan yang selalu membantu dan selalu direpotkan dalam setiap keadaan.
5. Teman seperjuangan satu Dosen Pembimbing dalam mengerjakan Tugas Akhir, Dinda, Agus, Sifa'ul, Diah dan Romli.
6. Mas Danang, Mas Habib, Uzu, Faiza, Lana yang membantu dalam pengerjaan Tugas Akhir ini.
7. Frikha, Ais, Bhara, Budi dan Didit yang telah meminjamkan peralatan yang menunjang dalam pengerjaan Tugas Akhir ini.
8. Dulur Matematika ITS 2013, Rozi, Prima, Ardi, Fadhlan, Sinar, Eries, Neni, Niken, Jessica, Melynda, Ayuk, Ayur, Lisa, Mega, Nurma, Diana, Yenny, Ina, Nastitie, Retno, Dinan, Via dan semua teman-teman yang tak bisa penulis sebutkan satu-persatu yang saling mendukung, menghibur dan memotivasi.
9. Teman seperjuangan SMA N 4 Bukittinggi yang diterima di ITS, Abidin dan Fajar.

10. TKO OMITS 2015, Keluarga SAINSTEK HIMATIKA ITS, IMAMI Surabaya, LDJ Ibnu Muqhlah Matematika ITS khususnya Departemen Dana Usaha.
11. Semua pihak yang tak bisa penulis sebutkan satu-persatu, terima kasih telah membantu sampai terselesaikannya Tugas Akhir ini.

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	iii
ABSTRAK	v
KATA PENGANTAR	xi
DAFTAR ISI	xv
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
BAB I PENDAHULUAN	
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah	3
1.4. Tujuan	3
1.5. Manfaat	4
1.6. Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA	
2.1. Penelitian Terdahulu	7
2.2. Pengertian Citra	7
2.3. Citra Digital	8
2.3.1. Digitalisasi Spasial (Sampling).....	8
2.3.2. Digitalisasi Intensitas (Kuantitas).....	9
2.4. Video Digital.....	10
2.5. Segmentasi Citra	10
2.6. Thresholding	11
2.7. Operasi Morfologi.....	12
2.7.1. <i>Structuring Element</i>	12
2.7.2. Dilasi.....	13
2.7.3. Erosi.....	13
2.7.4. Opening.....	14
2.7.5. Closing.....	14
2.8. <i>Noise</i> Pada Citra.....	14
2.9. <i>Boundary Object</i>	16
2.10. Metode Analisis <i>Blob</i>	16
2.11. Jarak Antara 2 Titik	17

BAB III METODOLOGI

3.1 Tahapan Penelitian	19
3.1.1. Studi Literatur dan Pengumpulan Data	19
3.1.2. Analisa dan Perancangan Program	19
3.1.3. Simulasi Program.....	19
3.1.4. Pengujian Program.....	19
3.1.5. Penarikan Kesimpulan	20
3.1.6. Pembuatan Laporan	20
3.2 Diagram Alir Penelitian	20

BAB IV PERANCANGAN DAN IMPLEMENTASI SISTEM

4.1. Analisa Sistem	23
4.1.1. Analisis Sistem Perangkat Lunak	23
4.1.2. Analisis Kebutuhan Sistem	28
4.2. Perancangan Sistem	28
4.2.1. Perancangan Data Sistem.....	28
4.2.2. Perancangan Proses.....	30
4.2.3. Perancangan Antar Muka Sistem.....	33
4.3. Implementasi Sistem	34
4.3.1. Implementasi Input Video.....	34
4.3.2. Implementasi Pemilihan area ROI	35
4.3.3. Implementasi Proses <i>grayscale</i>	36
4.3.4. Implementasi Proses Biner.....	36
4.3.5. Implementasi Proses Operasi Morfologi.....	37
4.3.6. Implementasi Deteksi Lampu Kendaraan	37
4.3.7. Implementasi <i>Pairing</i>	37
4.3.8. Implementasi Perhitungan Kendaraan	39

BAB V UJI COBA DAN PEMBAHASAN

5.1. Data Uji Coba.....	41
5.2. <i>Graphical User Interface (GUI)</i> Program.....	42
5.3. Hasil Proses Tahapan.	43
5.3.1. Proses <i>Pre-processing</i>	43
5.3.2. Proses Deteksi Bagian Lampu	49
5.3.3. Proses <i>Pairing</i>	50
5.3.4. Proses <i>Counting</i>	45
5.4. Uji Coba Video 1	51

5.4.1. Hasil proses <i>Pre-processing</i>	51
5.4.2. Hasil Deteksi dan <i>Pairing</i>	52
5.4.3. Hasil <i>Counting</i>	52
5.5. Uji Coba Video 2	54
5.5.1. Hasil proses <i>Pre-processing</i>	54
5.5.2. Hasil Deteksi dan <i>Pairing</i>	54
5.5.3. Hasil <i>Counting</i>	55
5.6. Uji Coba Video 3	46
5.6.1. Hasil proses <i>Pre-processing</i>	56
5.6.2. Hasil Deteksi dan <i>Pairing</i>	57
5.6.3. Hasil <i>Counting</i>	58
BAB VI PENUTUP	
6.1. Kesimpulan	61
6.2. Saran	61
DAFTAR PUSTAKA	63
BIODATA PENULIS	

DAFTAR GAMBAR

Gambar 2.1. Proses <i>sampling</i> citra dari citra analog ke citra digital	9
Gambar 2.2. Proses sampling dan kuantisasi.	9
Gambar 2.3. <i>Structuring Element</i>	13
Gambar 2.4. Noise pada Citra	15
Gambar 2.5. Analisis Blob	17
Gambar 2.6. Jarak Antara 2 titik	17
Gambar 3.1. Diagram Alir Penelitian.....	21
Gambar 3.2. Blok Diagram Proses Sistem Perangkat Lunak Deteksi dan Perhitungan Jumlah Kendaraan Bergerak Pada Waktu Malam Hari Berbasis Pengolahan Citra Digital.....	22
Gambar 4.1. Data Flow Diagram	24
Gambar 4.2. Proses <i>Scanning</i> Video	25
Gambar 4.3. Citra dengan ROI.....	26
Gambar 4.4. Proses <i>Grayscale</i>	26
Gambar 4.5. Proses Binerarisasi.....	27
Gambar 4.6. Penghilangan <i>Noise</i>	27
Gambar 4.7. Proses Deteksi Lampu Kendaraan	28
Gambar 4.8. Deteksi Kendaraan berdasarkan Lampu Depan.....	28
Gambar 4.9. <i>Layout</i> untuk Pengambilan Rekaman dengan Kondisi Malam hari	29
Gambar 4.10. <i>Data Flow Diagram</i> Sistem Perangkat Lunak Deteksi dan Perhitungan Jumlah Kendaraan Bergerak Pada Waktu Malam Hari Berbasis Pengolahan Citra Digital..	31
Gambar 4.11. <i>Data Flow Diagram</i> Proses <i>Pre-processing</i>	32
Gambar 4.12. Diagram Alir <i>Pairing</i>	33
Gambar 4.13. Tampilan Halaman Detail.....	34
Gambar 4.14. Antar Muka <i>Input</i> Video.....	36
Gambar 5.1. Tampilan Halaman Utama GUI.....	42
Gambar 5.2. Proses <i>Cropping</i> Citra	44
Gambar 5.3. Proses <i>Grayscale</i>	46

Gambar 5.4. Proses <i>Extended-Maxima Transform</i>	47
Gambar 5.5. Proses <i>Extended-Maxima Transform</i>	47
Gambar 5.6. Proses Biner	48
Gambar 5.7. <i>Structuring Element</i> Berbentuk <i>Disk</i>	48
Gambar 5.8. Proses <i>Filtering Noise</i>	49
Gambar 5.9. Proses <i>Bondaries</i>	49
Gambar 5.10. Proses Deteksi Lampu.	50
Gambar 5.11. Proses <i>Pairing</i>	51
Gambar 5.12. Proses <i>Counting</i>	52
Gambar 5.13. <i>Pre-processing</i> Video 1	53
Gambar 5.14. Hasil Deteksi dan <i>Pairing</i> Video 1	53
Gambar 5.15. Tampilan Akhir Hasil Video 1	54
Gambar 5.16. Proses <i>Pre-Processing</i> Video 2	55
Gambar 5.17. Hasil Deteksi dan <i>Pairing</i> Video 2	56
Gambar 5.18. Tampilan Akhir Hasil Video 2	56
Gambar 5.19. Proses <i>Pre-processing</i> Video 3.....	58
Gambar 5.20. Hasil Deteksi dan <i>Pairing</i> Video 3	58
Gambar 5.21. Tampilan Akhir Hasil Video 3	59

DAFTAR TABEL

Tabel 4.1. Tabel Data Proses.....	30
Tabel 5.1. Daftar Data Uji	41
Tabel 5.2. Akurasi Video 1	54
Tabel 5.3. Akurasi Video 2	56
Tabel 5.4. Akurasi Video 3	59

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Seiring dengan berkembangnya zaman, semua aspek kehidupan tidak lepas dari penggunaan teknologi. Salah satunya adalah penggunaan teknologi pada sektor transportasi. Pertumbuhan sektor transportasi di Indonesia sangat tinggi sehingga padatnya kendaraan yang berada di jalan raya. Berdasarkan data dari Portal Data Indonesia jumlah kendaraan bermotor pada tahun 2014 mencapai 114.209.266 unit kendaraan[1]. Dengan jumlah kendaraan yang semakin bertambah, fasilitas jalan saat ini tidak dapat mengimbangi sehingga terjadinya permasalahan-permasalahan lalu lintas.

Permasalahan lalu lintas yang sering terjadi adalah kemacetan yang harus diberi perhatian khusus oleh pemerintah dan masyarakat sekitar. Setiap orang mengharapkan transportasi yang dapat berjalan dengan cepat[2]. Tetapi disisi lain dengan banyaknya kendaraan di jalan raya yang menyebabkan kemacetan sehingga memperlambat perjalanan. Maka dari itu dibutuhkannya informasi kendaraan yang melintasi suatu jalan yang sesuai dengan luas jalan sehingga pembangunan infrastruktur jalan menjadi efektif dan mengurangi permasalahan lalu lintas yang ada.

Pengambilan data informasi kendaraan yang melintasi jalan raya dapat dilakukan dengan berbagai cara. Sebelumnya perhitungan kendaraan dilakukan secara manual dengan menggunakan kekuatan manusia[3]. Sekarang dengan berkembangnya teknologi, perhitungan kendaraan pada jalan raya dapat dilakukan dengan alat yang beroperasi secara *real time*[4]. *Intelligent Traffic System* (ITS) atau Sistem Transportasi Pintar yang merupakan penerapan teknologi, informasi dan komunikasi untuk sektor transportasi. ITS dapat memberikan manfaat pada sistem transportasi seperti mendapatkan informasi mengenai keadaan lalu lintas, mengurangi kemacetan dan antrian,

meningkatkan sarana dan prasarana transportasi, mengurangi polusi lingkungan dan permasalahan lalu lintas lainnya. Pengaplikasian ITS dalam sistem transportasi dapat dijadikan sebagai alat efektif sehingga dapat meningkatkan nilai mobilitas perjalanan dan mengurangi dampak permasalahan lalu lintas.

Pengambilan data secara *real time* harus menghadapi semua kondisi geografis yang terjadi pada saat pengambilan data seperti hujan, malam hari, bersalju dan kondisi alam lainnya[5]. Data informasi untuk sistem pemantauan kendaraan bergerak secara *real time* dapat diperoleh dari berbagai sumber seperti detektor lingkaran, detektor ultrasonik, sensor microwave, sensor radar atau kamera video[6]. Dengan adanya kemajuan teknologi terbaru yang berguna untuk teknik pengolahan citra. Kamera video yang telah ditemukan sebagai cara yang efisien untuk mengumpulkan dan menganalisis data lalu lintas dikarenakan sistem kamera berbasis video yang lebih canggih dan kuat karena informasi yang berhubungan dengan urutan gambar yang hadir dalam video memungkinkan untuk mengidentifikasi dan mengklasifikasikan kendaraan dengan cara yang paling efektif[6].

Teknik pengolahan video untuk mendapat data dan dapat mendeteksi kendaraan di jalan raya telah dilakukan beberapa peneliti. Salah satunya yaitu pada penelitian Zamroji Hariyanto yang dilakukan pada pagi dan siang hari[7]. Penelitian ini menggunakan data berupa video yang berdurasi 1 menit dengan keadaan jalan raya pada siang hari. Penelitian Zamroji H yang berjudul Klasifikasi Jenis Kendaraan Bergerak Berbasis Geometric Invariant Moment mengklasifikasikan kendaraan menjadi 3 jenis kendaraan yaitu motor, mobil dan truk.

Berbeda dengan kondisi pada siang hari, pada waktu malam hari juga merupakan kondisi dimana banyaknya terjadi permasalahan lalu lintas di jalan raya. Kondisi malam hari yang memiliki tingkat kontras cahaya yang rendah dan sensitivitas cahaya yang lemah sehingga sulit untuk mendeteksi kendaraan yang melintasi jalan raya yang berbeda dengan siang hari[8]. Selain itu terdapat juga refleksi dan iluminasi cahaya lampu yang

berasal pada lampu jalan dan kendaraan yang lain sehingga sulit untuk mendeteksi kendaraan. Dari penjabaran tersebut penulis memilih penelitian mengenai Deteksi dan Perhitungan Jumlah Kendaraan Bergerak pada Waktu Malam Hari.

1.2. Rumusan Masalah

Rumusan masalah dari Tugas Akhir ini adalah :

1. Bagaimana mendeteksi kendaraan yang melintasi jalan raya pada waktu malam hari berdasarkan lampu kendaraan?
2. Bagaimana menghitung jumlah kendaraan yang melintasi jalan raya pada waktu malam hari ?
3. Bagaimana membuat perangkat lunak yang mampu mengolah informasi dari video digital untuk mendeteksi dan menghitung jumlah kendaraan bergerak pada waktu malam hari ?

1.3. Batasan Masalah

Batasan masalah dari Tugas Akhir ini adalah :

1. Video yang diambil merupakan video keadaan jalan raya pada saat malam hari.
2. Video diambil dari atas dengan arah pergerakan objek hanya satu arah.
3. Kondisi cuaca pada saat pengambilan video dalam keadaan normal.
4. Informasi digital yang diproses dilakukan secara *offline*.
5. Kendaraan yang dideteksi adalah kendaraan roda 4 yang memiliki 2 lampu depan dalam keadaan yang baik.

1.4. Tujuan

Tujuan dalam usulan Tugas Akhir ini adalah :

1. Mendeteksi kendaraan yang bergerak yang melintasi jalan raya pada waktu malam hari berdasarkan lampu kendaraan.
2. Menghitung jumlah kendaraan bergerak yang melintasi jalan raya pada waktu malam hari.

3. Mengembangkan perangkat lunak yang mampu mengolah informasi dari video digital untuk mendeteksi dan menghitung jumlah kendaraan bergerak pada waktu malam hari.

1.5. Manfaat

Manfaat yang dapat diperoleh dari Tugas Akhir ini adalah Memberikan kontribusi pengembangan teknologi dalam bidang pengembangan perangkat lunak dan prototype untuk mendeteksi dan menghitung kendaraan bergerak pada waktu malam hari

1.6. Sistematika Penulisan

Sistematika Penulisan Tugas Akhir ini dibagi menjadi enam bab. Isi dari masing-masing bab sebagai berikut :

1. **BAB I PENDAHULUAN**

Pada bab ini dibahas tentang latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat dari penelitian yang dilakukan, serta sistematika penulisan Tugas Akhir.

2. **BAB II TINJAUAN PUSTAKA**

Pada bab ini diuraikan beberapa teori pendukung yang berasal dari jurnal dan buku yang berkaitan atau yang melandasi pembahasan pada penelitian ini untuk membantu menyelesaikan permasalahan Tugas Akhir.

3. **BAB III METODE PENELITIAN**

Pada bab ini berisi tentang langkah-langkah yang digunakan untuk menyelesaikan Tugas Akhir.

4. **BAB IV PERANCANGAN DAN IMPLEMENTASI SISTEM**

Pada bab ini akan menguraikan bagaimana tahapan tahapan dalam perancangan implementasi. Pembahasan perancangan implementasi dimulai dari perancangan pre-processing.

5. **BAB V PEMBAHASAN DAN UJI COBA**

Bab ini menyajikan pembahasan hasil uji coba program dari beberapa rekaman video arus lalu lintas pada malam hari. Kemudian dicatat sebagai bahan untuk merumuskan beberapa kesimpulan dan saran dari Tugas Akhir ini.

6. **BAB VI KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dari hasil penelitian yang telah dilakukan dan saran untuk pengembangan dari penelitian ini.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian tentang deteksi dan perhitungan jumlah kendaraan yang berada di jalan raya sebelumnya telah diteliti oleh beberapa peneliti yaitu Zamroji Hariyanto pada tahun 2014 melakukan penelitian mendeteksi dan mengklasifikasi kendaraan yang melewati jalan raya dengan menggunakan *Geometric Invariant Moment*. Dalam penelitiannya, perangkat lunak yang dirancang untuk mendeteksi kendaraan memiliki tingkat akurasi 91,81% dan terdeteksi 202 dari 220 kendaraan untuk pagi hari kemudian akurasi sebesar 94,79% dan terdeteksi 364 dari 284 kendaraan untuk siang hari[7]. Tetapi pada penelitian ini belum melakukan deteksi dan perhitungan kendaraan pada malam hari.

Selanjutnya yaitu penelitian Kostia Robert yang berjudul *Video-based traffic monitoring at day and night time*[2]. Pada penelitian yang dilakukan Kostia Robert yaitu membandingkan pendeteksian yang terjadi pada siang hari dan pada malam hari. Dari hasil penelitian ini menghasilkan tingkat akurasi pendeteksian sebesar 94,8% untuk sore hari dan 97% untuk malam hari. Tetapi untuk perhitungan belum dilakukan pada penelitian ini. Kemudian yaitu penelitian yang dilakukan oleh S. Padmavathi dkk yang berjudul *Vision based Vehicle Counting for Traffic Congestion Analysis during Night Time* yang melakukan perhitungan jumlah kendaraan di jalan raya pada malam hari[8].

2.2. Pengertian Citra

Pengertian Citra menurut Kamus Besar Bahasa Indonesia, Citra merupakan pemahaman kesan yang timbul karena pemahaman akan suatu kenyataan. Menurut G.Sach citra adalah pengetahuan mengenai kita dan sikap-sikap terhadap kita yang mempunyai kelompok-kelompok yang berbeda[11]. Citra terdiri dari dua jenis yaitu citra diam dan citra bergerak. Citra diam merupakan citra tunggal atau citra yang tidak bergerak.

Sedangkan citra bergerak yaitu rangkaian dari citra diam yang ditampilkan secara berurutan sehingga memberikan kesan bergerak pada saat dilihat dengan mata.

2.3. Citra Digital

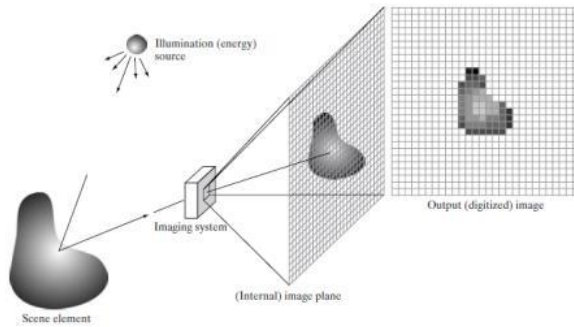
Citra digital merupakan merupakan fungsi dua dimensi yang dapat dinyatakan dengan fungsi $f(x,y)$, dimana x dan y merupakan titik koordinat spasial. Sedangkan amplitudo dari fungsi f pada sembarang koordinat (x,y) merupakan nilai intensitas cahaya, yang merupakan representasi dari warna cahaya yang ada pada citra analog, yang dijelaskan dalam dalam buku *Digital Image Processing* yang ditulis oleh Rafael. C. Gonzales dan Richard. E. Woods[9]. Citra digital adalah suatu citra dimana (x,y) dan nilai intensitas dari f terbatas (*discrete quantities*), dan telah dilakukan proses digitalisasi spasial dan digitalisasi kuantitas.

2.3.1. Digitalisasi Spasial (Sampling)

Sampling merupakan proses pengambilan informasi dari citra analog yang memiliki panjang dan lebar tertentu untuk membaginya ke beberapa blok kecil. Blok-blok tersebut disebut sebagai piksel. Sehingga citra digital yang lazim dinyatakan dalam bentuk matriks memiliki ukuran $M \times N$ dengan M sebagai baris dan N kolom. Bisa juga disebut sebagai citra digital yang memiliki $M \times N$ buah piksel. Notasi matriks citra digital dapat dinyatakan sebagai berikut :

$$f(x, y) = \begin{bmatrix} f(0,0) & \cdots & f(0, N-1) \\ \vdots & \ddots & \vdots \\ f(M-1,0) & \cdots & f(M-1, N-1) \end{bmatrix} \quad (1)$$

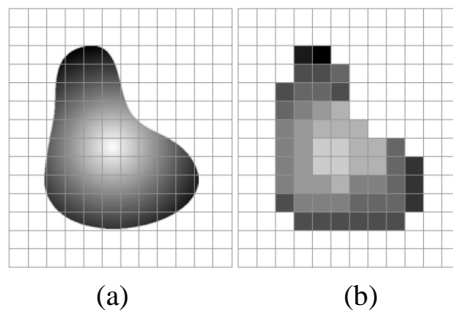
Proses *sampling* citra analog ke citra digital ditampilkan pada Gambar 2.1 di bawah ini :



Gambar 2.1. Proses *sampling* citra dari citra analog ke citra digital[9].
(sumber: “Digital Image Processing” [9])

2.3.2. Digitalisasi Intensitas (Kuantitas)

Kuantisasi adalah proses pemberian nilai derajat keabuan di setiap titik piksel yang merupakan representasi dari warna asli dari citra analog. Rentang nilai keabuan adalah 0 – 255.



Gambar 2.2. Proses sampling dan kuantisasi. (a) Gambar citra analog yang ditempatkan pada sensor *array*. (b) Gambar citra setelah proses *sampling* dan kuantisasi, tiap piksel pada citra *b* memiliki derajat keabuan masing-masing[9].
(sumber: “Digital Image Processing” [9])

2.4. Video Digital

Video adalah teknologi untuk menangkap, merekam, memproses, menyimpan, dan merekonstruksi suatu urutan dari beberapa citra. Alan C. Bovik menjelaskan bahwa video digital merupakan hasil *sampling* dan kuantisasi dari video analog[10]. Secara mendasar, tidak ada perbedaan proses *sampling* dan kuantisasi antara citra digital dan video digital.

Bagaimanapun juga, video analog yang kita lihat sehari-hari seperti tampilan pada TV analog, sebenarnya bukan sesuatu yang benar-benar kontinu, melainkan terdiri dari beberapa frame yang ditampilkan dengan kecepatan tertentu. Setiap *frame* merupakan citra analog dan kecepatan untuk menampilkan citra-citra yang ada disebut sebagai *frame rate* dengan satuan fps (*frame per second*). Jika *frame rate* cukup tinggi, maka akan terlihat sebagai rangkaian yang kontinu sehingga tercipta ilusi gerak yang halus.

Video analog dapat dinyatakan dengan fungsi $I(x,y,t)$, dimana (x,y) adalah nilai kontinu dari fungsi I dan t menyatakan waktu. Sebenarnya tampilan video analog di TV maupun monitor merupakan representasi dari fungsi sinyal elektrik satu dimensi $V(t)$ yang terdiri dari beberapa citra analog $I(x,y,t)$ dengan jumlah citra (x,y) tertentu dan waktu (t) tertentu. Proses pemisahan video ke beberapa unit frame citra disebut sebagai *scanning*[8].

2.5. Segmentasi Citra

Segmentasi adalah proses mempartisi citra digital menjadi beberapa segmen (set piksel, juga dikenal sebagai super pixels). Tujuan dari segmentasi adalah untuk menyederhanakan atau mengubah penyajian gambar ke sesuatu yang lebih bermakna dan lebih mudah untuk menganalisis. Gambar segmentasi biasanya digunakan untuk menemukan obyek dan batas-batas (garis, kurva, dll) dalam gambar. Lebih tepatnya, segmentasi citra adalah proses untuk menempatkan label untuk setiap pixel dalam sebuah gambar sehingga piksel dengan pangsa label yang sama karakteristik visual tertentu.

2.6. Thresholding

Thresholding adalah proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui daerah mana yang termasuk objek dan background dari citra secara jelas.

Beberapa jenis citra digital yang sering digunakan adalah citra biner, citra *grayscale* dan citra warna.

1. Citra Biner (Monokrom). Banyaknya dua warna, yaitu hitam dan putih. Dibutuhkan 1 bit di memori untuk menyimpan kedua warna ini.
2. Citra *Grayscale* (Skala Keabuan). Banyaknya warna tergantung pada jumlah bit yang disediakan di memori untuk menampung kebutuhan warna ini. Citra 2 bit mewakili 4 warna, citra 3 bit mewakili 8 warna, dan seterusnya. Semakin besar jumlah bit warna yang disediakan di memori, semakin halus gradasi warna yang terbentuk.
3. Citra Warna (*True Color*). Setiap piksel pada citra warna mewakili warna yang merupakan kombinasi dari tiga warna dasar (RG8 = *Red Green Blue*). Setiap warna dasar menggunakan penyimpanan 8 bit = 1 *byte*, yang berarti setiap warna mempunyai gradasi sebanyak 255 warna. Berarti setiap piksel mempunyai kombinasi warna sebanyak $28 \times 28 \times 28 = 224 = 16$ juta warna lebih. Itulah sebabnya format ini dinamakan *true color* karena mempunyai jumlah warna yang cukup besar sehingga bisa dikatakan hampir mencakup semua warna di alam.

Citra hasil *thresholding* biasanya digunakan lebih lanjut untuk proses pengenalan objek serta ekstraksi fitur. Metode *thresholding* secara umum dibagi menjadi dua, yaitu:

1. *Thresholding* global
Thresholding dilakukan dengan mempartisi histogram dengan menggunakan sebuah threshold (batas ambang) global T, yang berlaku untuk seluruh bagian pada citra.
2. *Thresholding* adaptif

Thresholding dilakukan dengan membagi citra menggunakan beberapa sub citra. Lalu pada setiap sub citra, segmentasi dilakukan dengan menggunakan threshold yang berbeda.

2.7. Operasi Morfologi

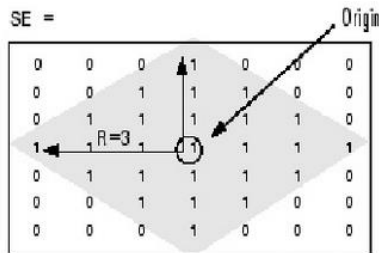
Morfologi citra merupakan suatu operasi pemrosesan citra yang mengolah citra berdasarkan bentuknya. Operasi morfologi mengaplikasikan suatu *Structuring Element* terhadap suatu citra masukan, membentuk suatu citra keluaran dengan ukuran yang sama dengan citra masukan. Pada operasi morfologi, nilai dari tiap piksel pada citra keluaran didasarkan pada perbandingan dari piksel pada citra masukan dengan piksel di sekitarnya[13].

Operasi morfologis yang paling dasar adalah operasi dilasi dan erosi. Dilasi menambahkan piksel pada boundaries dari suatu objek dari suatu citra, sedangkan erosi menghilangkan piksel dari suatu boundaries dari objek. Jumlah piksel yang ditambahkan atau dihilangkan tergantung dari ukuran dan bentuk dari *Structuring Element* yang digunakan untuk memproses citra.

2.7.1. Structuring Element

Structuring element merupakan suatu matriks yang hanya berisi dua nilai yaitu 1 atau 0 yang disusun sedemikian rupa sehingga membentuk suatu formasi dan ukuran, piksel dengan nilai 1 pada *structuring element* biasa disebut dengan neighborhood.

Structuring element dua dimensi memiliki ukuran yang biasanya jauh lebih kecil di banding dengan citra yang diolah. Piksel pada bagian tengah dari structuring element biasa disebut dengan origin.



Gambar 2.3. *Structuring Element*

(Sumber : Operasi Morfologi Citra Dengan Matlab[13].)

2.7.2. Dilasi

Pada dilasi, setiap background piksel yang menyentuh objek piksel dirubah menjadi objek piksel, hal tersebut membuat objek menjadi terlihat lebih gemuk. Operasi pada dilasi dilakukan dengan cara berikut, A dan B merupakan set pada Z^2 dan \emptyset menyatakan suatu set yang kosong, dilasi dari A oleh B dinyatakan dengan

$$A \otimes B \quad (2)$$

Didefinisikan sebagai

$$A \oplus B = \{x | (B')_x \cap A \neq \emptyset\} \quad (3)$$

B' adalah refleksi dari B, maka proses dilasi terdiri dari proses mencari refleksi dari B pada origin dan mengeser refleksi tersebut sebanyak x.

2.7.3. Erosi

Operasi erosi, sama seperti pada dilasi, dilakukan secara biner. Tetapi untuk erosi akan menghasilkan objek pada citra menjadi lebih tipis atau terkikis. Untuk membuat operasi erosi, kita tentukan E adalah suatu bidang Euclidean atau sebuah grid integer, dan A adalah suatu citra biner dalam himpunan E. Erosi dari suatu citra biner A dengan sebuah *structuring element* B.

$$A \ominus B = \{z | (B)_z \subseteq A\} \quad (4)$$

Dimana B_z adalah translasi dari B oleh vector z . Ketika *structuring element* B memiliki origin yang terletak di origin dari E , maka erosi dari A oleh B dapat diartikan sebagai lokus dari titik yang dicapai oleh origin dari B ketika B berada dalam A .

2.7.4. Opening

Opening merupakan suatu operasi morfologi citra berupa gabungan dari erosi dan dilasi, opening merupakan operasi dilasi terhadap erosi yang terjadi pada suatu set A dimana A adalah suatu citra masukan. Bersama dengan operasi closing, opening berperan dalam pemrosesan citra sebagai alat dasar untuk menghilangkan noise morfologi.

Opening menghilangkan objek-objek kecil dari foreground suatu citra. Opening biasa digunakan untuk mencari atau mendeteksi sesuatu yang besesuaian dengan suatu *structuring element*.

$$A \circ B = (A \ominus B) \oplus B \quad (5)$$

Rumusan dari operasi opening, terlihat bahwa set A mengalami operasi erosi terlebih dahulu, kemudian dilanjutkan dengan operasi dilasi.

2.7.5. Closing

Pada morfologi matematis, closing dari suatu set A oleh *structuring element* B adalah erosi dari dilatasi A oleh B .

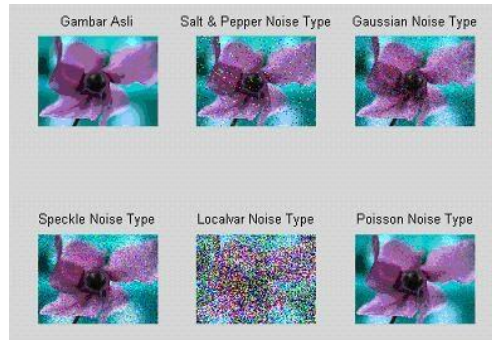
$$A \cdot B = (A \oplus B) \ominus B \quad (6)$$

Set A terlebih dahulu di dilasi oleh B , kemudian dilanjutkan dengan melakukan erosi ke proses tersebut. Closing akan menghilangkan lubang kecil pada citra sedangkan opening akan menghilangkan objek yang kecil.

2.8. Noise Pada Citra

Noise dapat diartikan sebagai gangguan atau kecatatan dari suatu citra yang tidak kita harapkan yang menyebabkan citra

menjadi rusak, tidak jelas, blur, bintik-bintik dan sebagainya. Ada banyak noise yang dapat mempengaruhi kualitas dari citra. Noise tersebut adalah



Gambar 2.4. Noise pada Citra

(sumber : <https://www.google.co.id/search?q=noise+pada+citra....>)

1. Salt and Peper

Pada citra akan nampak seperti titik-titik. Noise ini disebabkan oleh gangguan tajam atau tiba-tiba pada sinyal citra. Untuk citra RGB titik-titik muncul dalam tiga warna yakni merah (red), hijau (green) dan biru (blue), sedangkan pada citra gray noise akan muncul dalam dua warna yakni hitam (black) dan putih (white) yang tersebar pada citra. Noise ini memberikan efek "on dan off" pada pixel.

2. Gaussian

Disebut juga Gaussian White Noise. Noise ini dikatakan White Noise karena mempunyai distribusi normal. Nilai rata-rata dan variasinya besar maka citra seolah-olah hanya terlihat seperti citra putih saja.

3. Poisson

Poisson noise bukan merupakan noise buatan. Poisson merupakan noise yang ditambahkan langsung pada citra tanpa kita menambahkan parameter apapun, sehingga

efeknya pada citra pun tetap, berbeda dengan tipe noise yang sudah dijelaskan sebelumnya.

4. Speckle

Speckle merupakan noise ganda. Noise ini ditambahkan pada citra menggunakan persamaan $J=I+n*I$, dimana n terdistribusi random seragam dengan mean 0 dan variance V . V adalah konstanta non negative yang besarnya dapat berubah-ubah. Default nilai untuk V adalah 0.04. Makin besar nilai V maka citra akan semakin kabur. Noise speckle sering dijumpai pada aplikasi radar.

5. Localvar

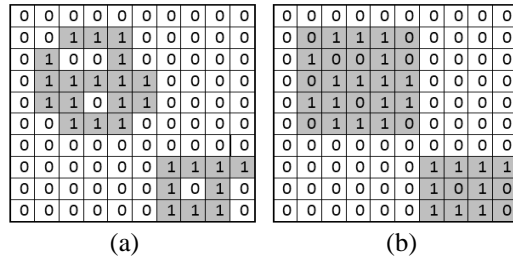
Localvar merupakan Gaussian noise dengan mean 0, dengan variance noise adalah fungsi dari intensitas citra yang nilainya berada dalam matrik citra. Vektor intensitas citra tidak boleh bernilai sama karena citra akan nampak sebagai layar putih.

2.9. *Boundary Object*

Pendeteksian kontur merupakan langkah awal membentuk boundary objek pada sebuah citra digital. Kontur adalah rangkaian piksel-piksel tepi yang membentuk batas daerah (region boundary). Batas daerah berguna untuk mendeskripsikan bentuk objek dalam tahapan analisis citra seperti pengenalan objek. Melalui bentuk kontur sehingga mampu mendeteksi tepi-tepi objek di dalam citra[14].

2.10. *Metode Analisis Blob*

Analisis *blob* ini menggunakan metode *connected component*, dimana di setiap kumpulan piksel yang tingkat keabuaannya bernilai satu, dikategorikan sebagai satu objek. Setiap objek yang terdeteksi akan diberi label untuk mempermudah pengolahan. Pada Gambar 2.3 menjelaskan tentang pemilihan area *blob* yang terdeteksi. Dari analisis *blob* tersebut akan diperoleh informasi tentang *centroid*, luas area, tinggi, dan lebar sebuah objek dari bentuk *rectangle*.



Gambar 2.5. Analisis Blob. Bagian (a) adalah representasi citra ukuran 10x10 piksel dan citra dengan nilai piksel 1 merupakan representasi dari objek (*foreground*). Sedangkan bagian (b) merupakan *blob* yang terdeteksi.

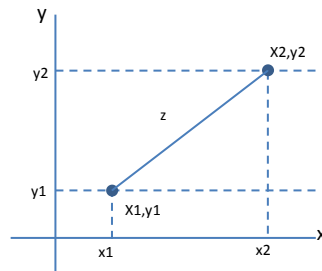
(sumber: “Penghitungan Kendaraan Bergerak Berbasis Algoritma Background Subtraction Menggunakan Metode Gaussian Mixture Model” [7])

2.11. Jarak Antara 2 Titik

Perhitungan jarak antara 2 titik dilakukan dengan menggunakan rumus *phytagoras* seperti pada dibawah ini

$$z = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (7)$$

dengan (x_1, y_1) merupakan koordinat titik pertama dan (x_2, y_2) merupakan koordinat titik ke dua. Jika digambar pada koordinat kartesian maka menjadi seperti pada gambar 2.6 berikut.



Gambar 2.6. Jarak antara 2 titik

BAB III METODOLOGI

Pada bab ini dijelaskan langkah-langkah yang digunakan dalam penyusunan Tugas Akhir. Disamping itu, dijelaskan pula prosedur dan proses pelaksanaan tiap-tiap langkah yang dilakukan dalam menyelesaikan Tugas Akhir.

3.1. Tahapan Penelitian

3.1.1. Studi Literatur dan Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data yang diperlukan pada penulisan tugas akhir ini. Data yang dikumpulkan berupa video keadaan jalan raya pada waktu malam hari. Video diambil dari atas jembatan penyeberangan dengan arah kendaraan satu arah. Kemudian melakukan penelitian *Video Processing* yang diperlukan sebagai metode dalam menyelesaikan permasalahan pada Tugas Akhir ini dengan cara mempelajari literatur-literatur ilmiah yang memiliki hubungan dengan topik penelitian yang sedang penulis lakukan guna mendapat data teoritis sehubungan dengan pembahasan penelitian ini.

3.1.2. Analisa dan Perancangan Program

Pada tahap ini akan dilakukan perancangan program berupa memilih dan menganalisis video yang telah didapatkan. Kemudian akan digunakan teknik pengolahan citra digital yaitu deteksi tepi untuk menentukan obyek kendaraan. Kemudian mengubah citra menjadi citra biner dan dilakukan operasi morfologi citra pada video yang telah didapatkan untuk mendapatkan obyek yang diperlukan pada tugas akhir ini.

3.1.3. Simulasi Program

Pada tahap ini dilakukan pembuatan implementasi dari tugas akhir ini berupa simulasi program sesuai dengan perancangan program yang telah dilakukan dengan menggunakan bahasa pemrograman Matlab.

3.1.4. Pengujian Program

Pada tahap ini dilakukan uji coba simulasi perogram yang telah dibuat. Setelah itu akan dilihat apakah pada video yang didapatkan dapat mendeteksi kendaraan yang lewat dan apakah dapat menghitung jumlah kendaraan yang melewati jalan tersebut. Pada penelitian ini akan dilakukan pendeteksian dan perhitungan kendaraan berdasarkan lampu kendaraannya. Apabila terjadi kesalahan dan error, akan dilakukan perbaikan program sehingga didapatkan hasil yang baik, akurat dan sesuai dengan yang diinginkan.

3.1.5. Penarikan Kesimpulan

Pada tahap ini dilakukan penarikan kesimpulan terhadap dari hasil pengerjaan, uji coba yang telah dilakukan dan pembahasan dari pengerjaan Tugas Akhir ini serta disampaikan saran untuk pengembangan berikutnya.

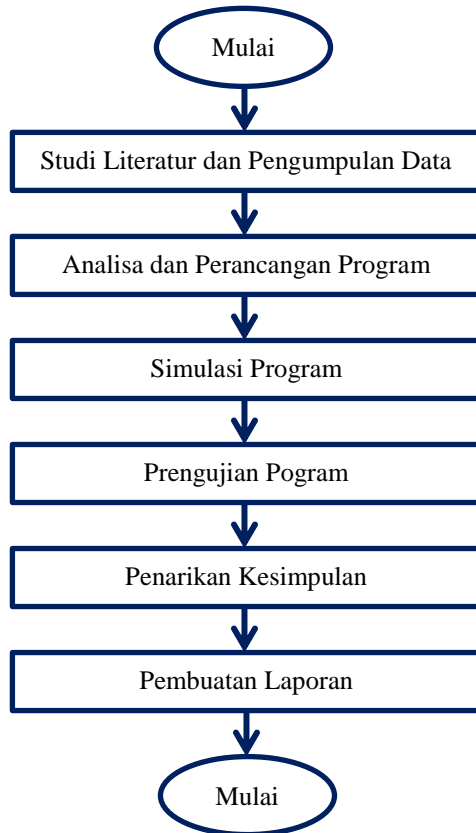
3.1.6. Pembuatan Laporan

Bagian terakhir dalam Tugas Akhir ini adalah membuat laporan seluruh tahapan/proses yang sudah dilakukan.

3.2. Diagram Alir Penelitian

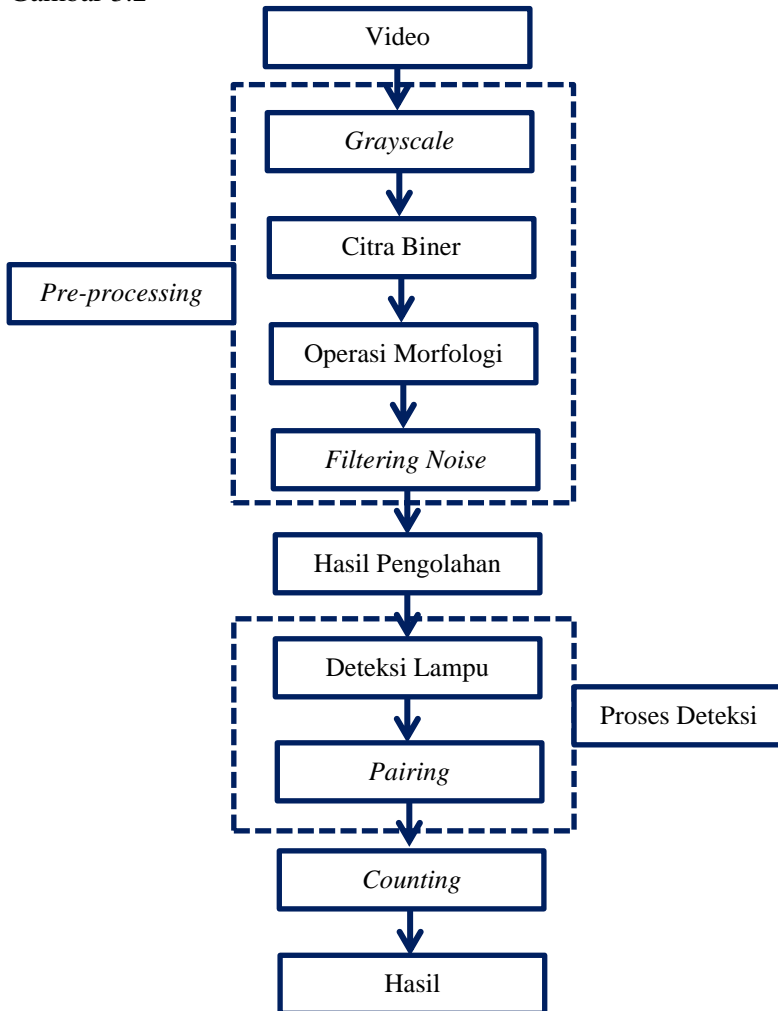
Berdasarkan uraian tersebut diatas, penelitian Tugas Akhir ini dapat dinyatakan dalam diagram alir sebagai berikut.

Gambar 3.1 menunjukkan diagram alir pembuatan tugas akhir ini.



Gambar 3.1. Diagram Alir Penelitian

Alur proses pembuatan perangkat lunak dapat dilihat pada Gambar 3.2



Gambar 3.2. Blok Diagram Proses Perangkat Lunak Deteksi Dan Perhitungan Jumlah Kendaraan Bergerak Pada Waktu Malam Hari Berbasis Pengolahan Citra Digital

BAB IV

PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab ini akan dibahas mengenai perancangan dan implementasi sistem dimulai dari pembahasan proses pengambilan data masukan, pengolahan data masukan dengan menggunakan pengolahan citra digital serta penjelasan mengenai cara untuk mendapatkan data keluaran yang sesuai dengan tujuan dari penelitian Tugas Akhir ini.

Selain hal yang telah disebutkan di atas, perancangan sistem juga meliputi perancangan antar muka (*user interface*) untuk memudahkan penilitan dalam memproses data dan melakukan analisis terhadap keluaran dari sistem yang telah dibangun.

Hasil dari analisis dan perancangan sistem dilanjutkan dengan implementasi sistem. Sehingga perangkat lunak yang dibutuhkan untuk memproses informasi dari rekaman video digital dapat dilakukan.

4.1. Analisa Sistem

Untuk membangun perangkat lunak yang dapat mendeteksi dan menghitung jumlah kendaraan bergerak pada waktu malam hari berbasis video digital. Maka diperlukan sistem yang dapat mengolah data masukan berupa rekaman arus kendaraan, pada kasus Tugas Akhir ini menggunakan data masukan berupa video digital arus kendaraan satu arah secara *offline*. Dengan menggunakan Pengolahan Citra Digital dan melakukan Pengolahan Video pada data yang didapatkan untuk dapat mendeteksi objek kendaraan serta dapat menghitung jumlah kendaraan yang lewat sehingga tercipta sistem yang sesuai dengan tujuan Tugas Akhir ini.

4.1.1. Analisis Sistem Perangkat Lunak

Perangkat lunak yang akan dibangun dapat digunakan oleh dinas perhubungan untuk mengamati kondisi arus lalu lintas di suatu jalan sehingga dapat menganalisa kebutuhan infrastruktur

jalan tersebut. Selain itu masyarakat juga dapat memperoleh informasi kondisi lalu lintas pada suatu jalan. Untuk proses alur dari tugas akhir ini ditunjukkan pada *Data Flow Diagram* pada sistem perangkat lunak Deteksi dan Perhitungan Jumlah Kendaraan Bergerak Pada Waktu Malam Hari yang ditunjukkan pada gambar 4.1.

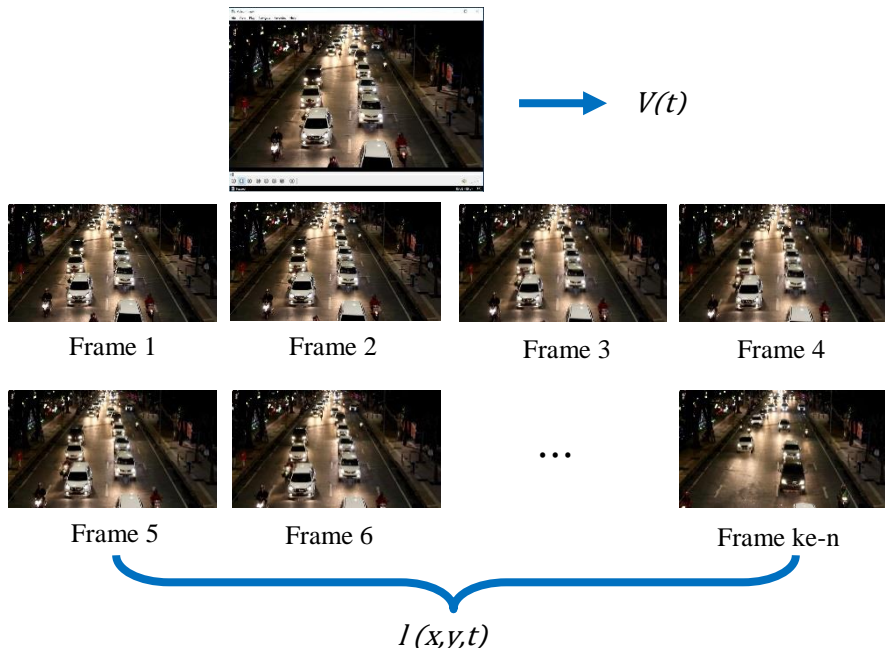


Gambar 4.1. *Data Flow Diagram*

Sistem perangkat lunak yang dibangun ini memiliki beberapa tahapan sebagai berikut :

a. Akuisi Video

Data masukan berupa rekaman video digital *offline* arus kendaraan yang kemudian di-*scanning*. *Scanning* adalah proses pemecahan video menjadi beberapa rangkaian citra yang sering disebut dengan *frame*.



Gambar 4.2. Proses *Scanning Video*

Pengambilan video arus lalu lintas kendaraan di beberapa jalan di daerah Surabaya. Video diambil menggunakan kamera digital Canon EOS 10d dan Canon 600D. Video yang diambil berdasarkan ketentuan berikut:

- Sudut pandang yang diambil ketika merekam adalah dari atas dengan jangkauan pandangan seluruh jalan satu arah.
- Video diambil dalam keadaan malam hari.

b. Penentuan ROI (*Region Of Interest*)

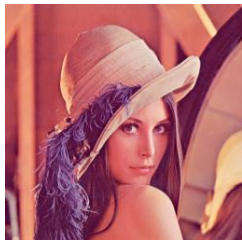
ROI (*Region of Interest*) akan digunakan sebagai area pendeteksian kendaraan bergerak.



Gambar 4.3. Citra dengan ROI (*Region of Interest*)

c. *Grayscale*

Pada sistem ini akan membagi video menjadi beberapa frame. Segmentasi citra yang digunakan pada penelitian ini adalah pertama mengubah citra RGB ke dalam bentuk grayscale yang dihasilkan seperti pada Gambar 4.4.



(a)



(b)

Gambar 4.4. Proses *Grayscale*, (a) Citra RGB, (b) Citra grayscale ,
(sumber : <https://www.google.co.id/search?q=citra+lena...>)

d. Binerarisasi

Kemudian dilanjut dengan melakukan proses thresholding pada citra yang telah dirubah menjadi citra grayscale untuk mendapatkan citra biner. Gambar 4.5 menunjukkan hasil perubahan citra *grayscale* menjadi citra biner.



Gambar 4.5. Proses Binerarisasi, (a) Citra *Grayscale*, (b) Citra Biner ,
(sumber : <https://www.google.co.id/search?q=citra+lena...>)

e. Operasi Morfologi dan *Filtering Noise*

Morfologi citra merupakan suatu operasi pemrosesan citra yang mengolah citra berdasarkan bentuknya . Pada citra biner yang telah didapatkan, dilakukan operasi morfologi opening yang sekaligus berfungsi untuk menghilangkan gangguan (*noise*) yang terdapat pada citra yang akan diteliti sehingga didapatkan hasil citra yang memiliki noise yang sedikit atau tidak ada sama sekali *noise* yang memberikan hasil seperti pada Gambar 4.6.



Gambar 4.6. Penghilangan Noise (a) Citra dengan noise, (b) Citra dengan noise yang telah hilang
(sumber : <https://www.google.co.id/search?q=noise+pada+citra....>)

f. Deteksi objek lampu kendaraan

Pendeteksian objek pada sistem ini terdapat pada lampu depan kendaraannya. Setelah melakukan proses segmentasi

sehingga mendapatkan yang mana bagian yang merupakan lampu dari kendaraan tersebut. Dari hasil segmentasi, kemudian pendeteksian bagian dari lampu seperti pada gambar berikut.



(a)



(b)



(c)

Gambar 4.7. Proses Deteksi Lampu Kendaraan, (a) Citra Biner (b) Penghilangan noise (c) Lampu kendaraan

g. Deteksi Kendaraan

Setelah didapatkannya bagian lampu dari kendaraan. Maka selanjutnya akan dilakukan pendeteksian kendaraan berdasarkan lampu pada kendaraan. Sehingga didapatkan hasil seperti pada gambar berikut.



Gambar 4.8. Deteksi kendaraan berdasarkan lampu depan

4.1.2. Analisis Kebutuhan Sistem

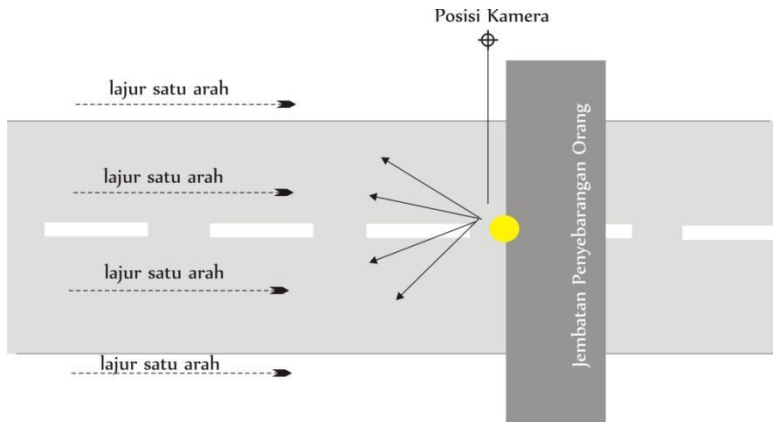
Perangkat lunak ini dibangun menggunakan bahasa pemrograman MATLAB R2016a. Selain itu juga diperlukan kamera digital untuk merekam arus kendaraan di jalan. Program ini dikembangkan menggunakan komputer dengan spesifikasi memiliki RAM 4 GB dengan *processor* intel core-i3 CPU.

4.2. Perancangan Sistem

4.2.1. Perancangan Data Sistem

a. Data Masukan

Data masukan sistem ini berupa video *offline* rekaman kendaraan di jalan yang diambil menggunakan kamera digital. Video diambil dengan posisi lensa kamera menjangkau ke seluruh area jalan satu lajur. Kamera diposisikan berada diatas jembatan penyebrangan yang merekam lajur satu arah secara keseluruhan. *Layout* untuk pengambilan video rekaman dapat dilihat pada Gambar 4.9



Gambar 4.9. *Layout* untuk pengambilan rekaman dengan kondisi malam hari [7]

File video yang diambil memiliki spesifikasi sesuai dengan batasan masalah yang terdapat dalam penelitian Tugas Akhir ini yaitu :

- Format ekstensi rekaman video offline berupa *mp4.
- Memiliki kemampuan merekam video dengan kecepatan minimal 25 *frame* per-second.
- Memiliki kemampuan merekam video dengan resolusi 320 x 240 piksel citra.

b. Data Proses

Data proses merupakan data yang digunakan dalam proses pengolahan data masukan. Data proses ini diperoleh dari hasil pengolahan data masukan sesuai dengan tahapan algoritma dan metode yang telah disusun. Tabel 4.1 menjelaskan tahapan dari data proses.

Tabel 4.1. Tabel Data Proses

No	Tahapan	Input	Output
1	Inputan awal	Video	Frame Citra
2.	Area ROI	Frame Citra	Citra ROI
3	<i>Grayscale</i>	Video	Citra <i>Grayscale</i>
4	Binerization	Citra <i>Grayscale</i>	Citra hitam putih (biner)
5	Operasi Morfologi	Citra hitam putih (biner)	Citra Operasi morfologi
6	Penghilangan Noise	Citra Operasi morfologi	Citra biner yang noise telah hilang
7	Deteksi bagian Lampu	Citra biner yang noise telah hilang	Citra yang lampu kendaraan telah terdeteksi
8	Pairing	Citra yang lampu kendaraan telah terdeteksi	Citra hasil pairing
9	Counting	Citra hasil pairing	Counting kendaraan

c. Data Keluaran

Data keluaran merupakan hasil deteksi kendaraan pada waktu malam hari serta hasil perhitungan jumlah kendaraan yang lewat. Program ini juga menghasilkan keluaran berupa ROI dan video yang menampilkan *pairing* dan *counting*.

4.2.2. Perancangan Proses

a. Perancangan Proses Algoritma

Proses algoritma ini dimulai dengan proses segmentasi citra yaitu proses *grayscale*, citra biner, filter noise dan operasi morfologi. Setelah melakukan proses segmentasi, selanjutnya masuk ke proses pendeteksian bagian lampu kendaraan dan proses *pairing* untuk kendaraan roda 4.

Gambar 4.10. menjelaskan alur sistem perangkat lunak Deteksi dan Perhitungan Jumlah Kendaraan Bergerak Pada Waktu Malam Hari.

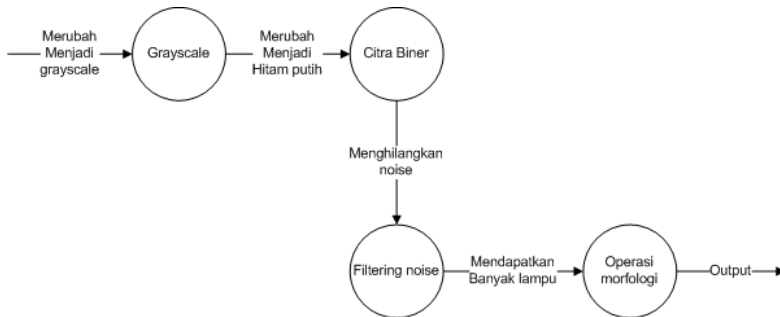


Gambar 4.10. Data Flow Diagram Sistem Perangkat Lunak Deteksi dan Perhitungan Jumlah Kendaraan Bergerak Pada Waktu Malam Hari

b. Perancangan Proses *Pre-processing* dan Deteksi Lampu

Pada proses *pre-processing* terdapat perubahan bentuk citra yaitu perubahan citra RGB menjadi citra *grayscale*. Setelah mendapatkan citra *grayscale* kemudian mengubah kedalam bentuk citra biner. Pada citra biner dilakukan operasi morfologi citra. Kemudian dilakukan pengurangan *noise* atau *filtering*. Hasil dari *filtering noise* didapatkan boundary yang akan digunakan untuk proses deteksi lampu kendaraan.

Untuk proses *Pre-processing* secara terurut dapat dilihat pada gambar 4.11.

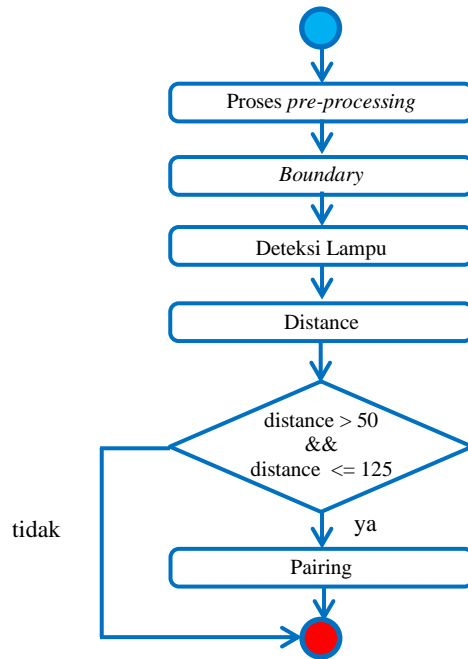


Gambar 4.11. *Data Flow Diagram Proses Pre-processing*

c. Perancangan Proses *Pairing*

Setelah mendapatkan bagian lampu kendaraan, masuk ke proses *pairing* yang berfungsi untuk memasangkan lampu kendaraan untuk kendaraan roda 4 yang memiliki 2 lampu. Sehingga nantinya pada saat proses perhitungan, lampu kendaraan yang pada roda 4 yang memiliki 2 lampu akan dihitung sebagai satu objek.

Gambar 4.12 menjelaskan alur tahapan pembentukan dari proses *pairing*.



Gambar 4.12. Diagram Alir Proses *Pairing*

Pada proses *pairing* akan digunakan *distance* (terhadap pixel) atau jarak dari setiap blob / lampu yang terdeteksi. Disini akan dipilih jarak yang memenuhi sehingga memberikan hasil yang baik. Untuk mencari jarak antara satu lampu dengan yang lain digunakan persamaan seperti dibawah

$$distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

dimana x_1 dan x_2 adalah posisi lampu terhadap sumbu x dan y_1 dan y_2 posisi lampu terhadap sumbu y.

d. Perancangan Proses *Counting*

Setelah mendapatkan bagian lampu kendaraan. Maka akan dilakukan proses perhitungan jumlah kendaraan di jalan raya pada malam hari berdasarkan lampu kendaraan yang didapat. Proses perhitungan dimulai ketika lampu kendaraan telah melewati area ROI yang telah di tentukan sebelumnya. Jika telah melewati area ROI maka perhitungan akan terus bertambah sampai video selesai dijalankan.

4.2.3. Perancangan Antar Muka Sistem

Desain antar muka sistem dibutuhkan agar pengguna dengan mudah mengoperasikan perangkat lunak yang dibangun. Desain ini dibuat semenarik mungkin dan *user friendly*.

Pada halaman detail ini ditampilkan video *input*, video pada daerah ROI, *counting* dan berjalan saat frame ke-. Desain antar muka dari halaman detail ditunjukkan pada Gambar. 4.13



Gambar 4.13.Tampilan Halaman Detail

4.3. Implementasi Sistem

4.3.1. Implementasi *Input* Video

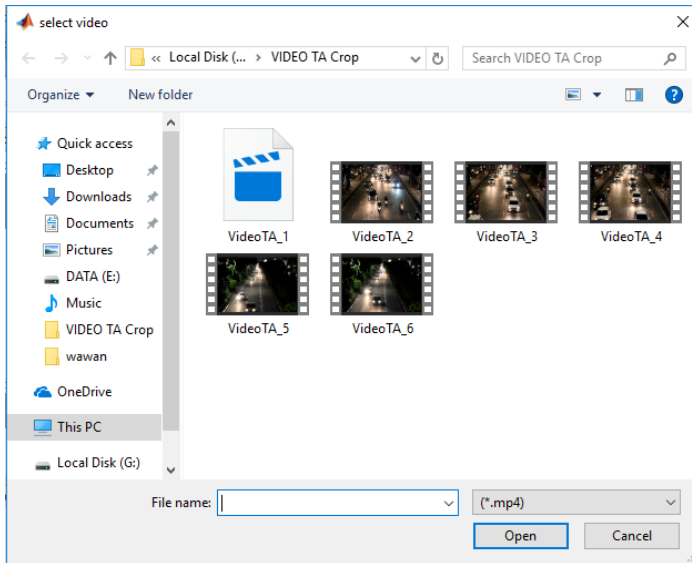
Pada proses ini akan ditentukan video yang nantinya digunakan sebagai data masukan untuk diproses. Penjabaran tentang proses pemilihan video adalah sebagai berikut :

Fungsi : menginputkan video bertipe *.mp4 pada form simulasi
 Input : video bertipe *.mp4
 Deskripsi : mengambil video bertipe *.mp4 yang telah disimpan pada computer

Gambar 4.14 adalah tampilan antar muka pengambilan *input* video. Kode program untuk menerima *input* video adalah sebagai berikut :

```
[FileName,PathName] = uigetfile({'*.mp4','
(*.mp4) MP4 video file'},...'Select input
MP4 video...');
if FileName~=0
    set(handles.Direktori,'String',strcat(Pa
thName,FileName));
    Direktori_Callback(handles.Direktori,[],
handles);
    vid = fullfile(PathName, Filename);
end
```

Untuk kode program lengkap, dapat dilihat pada lampiran



Gambar 4.14. Antar muka *input* video

4.3.2. Implementasi Pemilihan area ROI

Region of Interest (ROI) adalah daerah bagian dari citra atau *frame* yang akan diproses pada sistem ini. Pemilihan ROI ini bertujuan untuk memudahkan dan membantu program dalam menspesifikasikan kebutuhan program. Ukuran ROI pada sistem ini dianjurkan memuat setengah dari ukuran *frame* inputan. Sehingga dapat menangkap objek kendaraan secara utuh dan hasil yang diinginkan lebih akurat dan lebih efisien. Hal ini dikarenakan program hanya mengolah informasi dari bagian piksel yang dibutuhkan saja yaitu ROI. Kode program untuk pemilihan ROI adalah sebagai berikut :

```
h = imrect;
setColor(h, 'red');
position = wait(h);
```

4.3.3. Implementasi Proses *grayscale*

Citra video yang telah diinput akan memasuki proses yang pertama yaitu pengubahan citra video yang awalnya adalah citra dalam bentuk RGB menjadi citra *grayscale* (abu-abu). Berikut adalah penjabaran proses pengubahan citra RGB menjadi citra *grayscale*.

```
image = read(reader,i);
frameke = frameke + 1;
set(handles.frameKe, 'String', frameke);
% Proses grayscale
r = image(:, :, 1);
g = image(:, :, 2);
b = image(:, :, 3);
gray = 0.21*r + 0.72*g + 0.07*b;
```

4.3.4. Implementasi proses biner

Setelah didapatkan citra *grayscale*, kemudian masuk ke proses pengubahan citra *grayscale* menjadi citra biner (hitam putih). Pada proses biner disini menggunakan extended-maxima transform. Kemudian dengan citra biner tersebut akan dilakukan filtering untuk menghilangkan noise yang terdapat pada citra biner sehingga mendapatka citra yang dibutuhkan.

```
biner = imextendedmax(gray,1);
```

4.3.5. Implementasi Operasi Morfologi *Filtering Noise*

Operasi morfologi dilakukan setelah mendapatkan hasil dari citra yang telah dilakukan filter noise. Blob yang terdeteksi diproses Operasi Morfologi yang nantinya akan digunakan sebagai objek untuk mendeteksi lampu kendaraan.

```
b = strel('disk',7);
biner2 = imopen(biner,b);
```

4.3.6. Implementasi Deteksi lampu kendaraan

Pada tahap ini akan dilakukan proses operasi morfologi untuk mendapatkan lingkaran dari lampu kendaraan. Setelah didapatkan lingkaran dari lampu kendaraan, maka akan dilakukan pengubahan warna dari lingkaran tersebut kemudian akan ditampilkan pada video yang menunjukkan bahwa bagian tersebut adalah lampu kendaraan.

```
boundaries = bwboundaries(biner2);
numberOfBoundaries = size(boundaries(),1);
for k = 1 : numberOfBoundaries
    thisBoundary = boundaries{k};
    plot(thisBoundary(:,2),
         thisBoundary(:,1), 'r', 'LineWidth', 3);
end
```

4.3.7. Implementasi *pairing*

Untuk melakukan *pairing* diperlukannya titik tengah dari lingkaran lampu kendaraan. Dengan didapatkannya titik tengah dari lampu kendaraan yang kemudian akan digunakan untuk proses *pairing*. Lampu kendaraan roda 4 memiliki lampu kendaraan yang sejajar secara horizontal. Untuk melakukan *pairing* dapat dilakukan dengan cara menghubungkan lampu dengan lampu tetangga dengan jarak terdekat dari kiri ke kanan.

```
Boundaries = bwboundaries(biner2);
numberOfBoundaries = size(boundaries,1);
for b1 = 1 : numberOfBoundaries
    for b2 = 1 : numberOfBoundaries
        if b1 == b2
            continue;
        end
        boundary1 = boundaries{b1};
        boundary2 = boundaries{b2};
        boundary1x = boundary1(:,2);
        boundary1y = boundary1(:,1);
        x1 = 1;
```

```

        y1 = 1;
        x2 = 1;
        y2 = 1;
        overallMinDistance = inf;
        for k = 1 : size(boundary2, 1)
            boundary2x = boundary2(k,2);
            boundary2y = boundary2(k,1);
            allDistances = sqrt((boundary1x -
boundary2x).^2 + (boundary1y -
boundary2y).^2);
            [minDistance(k), indexOfMin] =
min(allDistances);
            if minDistance(k) <
overallMinDistance
                x1 = boundary1x(indexOfMin);
                y1 = boundary1y(indexOfMin);
                x2 = boundary2x;
                y2 = boundary2y;
                overallMinDistance =
                    minDistance(k);
            end
        end
        minDistance = min(minDistance);
        if minDistance < 125
            if minDistance > 50
                minDistance
                line([x2,x1],[y1,y2],'Color','
c','LineWidth',2);
            end
        end
    end
end
end

```

4.3.8. Implementasi Perhitungan Kendaraan

Untuk proses perhitungan (*counting*) dilakukan dengan cara menghitung daerah lampu (*blob*) yang terdeteksi yang telah melakukan proses *pairing*. Sehingga kendaraan yang telah terdeteksi dapat dihitung.

```
Iarea = bwareaopen(Morpho,1300);  
Idiff = Morpho - Iarea;  
Ifinal = bwareaopen(Idiff,10);  
  
stat = regionprops(Ifinal);  
str = sprintf('number of detected lamp : %d',  
floor(length(stat)/2));  
if str ~= 0  
    hitung=str;  
else  
    jumlah=hitung+jumlah;  
end
```

BAB V

UJI COBA DAN PEMBAHASAN

Bab ini menjelaskan mengenai pengujian program dan pembahasan dari hasil uji coba. Pengujian yang dilakukan disini adalah pengujian program dengan *input* video rekaman kendaraan bergerak di jalan raya. Pengambilan video dilakukan pada waktu malam hari dengan waktu yang berbeda-beda.

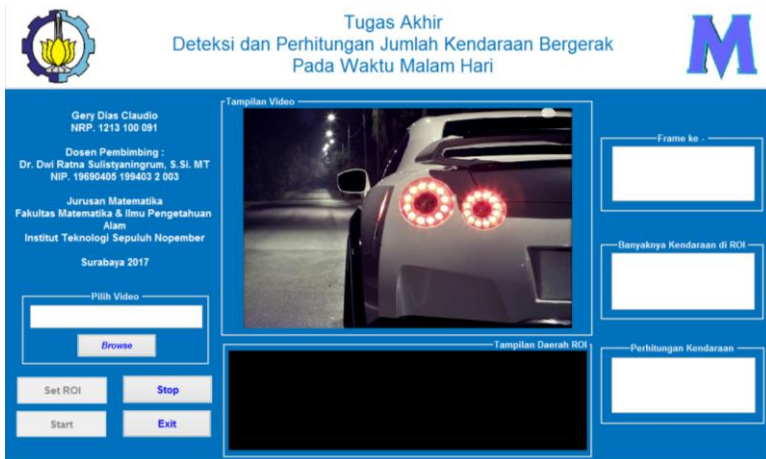
5.1. Data Uji Coba

Uji coba pada program dalam Tugas Akhir ini dilakukan terhadap video *.mp4. Video uji coba sudah tersimpan dalam penyimpanan komputer dan diperoleh dari hasil rekaman. Video yang digunakan memiliki pixel 1280 X 720 dengan frame rate minimal sebesar 25 frame/detik. Daftar input uji coba tersebut antara lain disajikan dalam Tabel 5.1.

Tabel. 5.1. Daftar Data Uji

No	Nama Video	Durasi	Tempat
1.	Video 1	30 detik	Jembatan penyebrangan di Jalan Kenjeran.
2.	Video 2	32 detik	Jembatan penyebrangan di Surabaya Plaza, Embong Kaliasin, Genteng Surabaya
3.	Video 3	24 detik	Jembatan jalan tol Surabaya-Porong yang terdapat di Jalan Manunggal Kebonsari, Kebonsari, Jambangan, Surabaya

5.2. Graphical User Interface (GUI) Program



Gambar 5.1. Tampilan Halaman Utama GUI

Halaman utama GUI ini memiliki desain antar muka yang ditunjukkan pada Gambar 5.1. Pada halaman ini dilakukan proses pemilihan video dan pemilihan ROI. Halaman ini menampilkan video hasil deteksi kendaraan. Antar muka halaman utama terdiri dari :

- **Axis 1**, berfungsi sebagai menampilkan video yang diinput, memilih area ROI beserta video yang telah diproses.
- **Axis 5**, berfungsi untuk menampilkan video yang terdapat pada daerah ROI
- **Push Button Browse**, berfungsi untuk memilih video input yang akan ditampilkan yang akan digunakan proses deteksi dan perhitungan.
- **Push Button Set ROI**, berfungsi untuk menentukan dan memilih area ROI.
- **Push Button Start**, berfungsi untuk menjalankan video yang telah dipilih.

- *Push Button Stop*, berfungsi untuk memberhentikan video yang telah dijalankan.
- *Push Button Close*, berfungsi untuk menutup halaman utama.
- *Edit Text Browse*, berfungsi untuk menampilkan alamat video yang diinput.
- *Edit Text Frame ke-*, berfungsi untuk menampilkan frame ke berapa dari video.
- *Edit Text CountROI*, berfungsi untuk menampilkan jumlah kendaraan yang terdeteksi pada area ROI.
- *Edit Text Counting*, berfungsi untuk menampilkan jumlah kendaraan yang terdeteksi secara keseluruhan.

5.3. Hasil Proses Tahapan

5.3.1. Proses *Preprocessing*

a. Proses *grayscale*

Citra video yang telah diinput akan memasuki proses yang pertama yaitu pengubahan citra video yang awalnya adalah citra dalam bentuk RGB menjadi citra *grayscale* (abu-abu).

Untuk mendapatkan citra *grayscale* pada citra RGB yang akan dirubah harus melalui proses pengubahan nilai dari citra RGB tersebut. Dikarenakan nilai dari suatu citra memiliki nilai matriks yang sangat besar maka untuk menunjukkan perhitungan pembentukan citra *grayscale*, maka dilakukan *cropping* terhadap citra RGB. Pada Gambar 5.2 menunjukkann citra RGB dan citra yang telah di *cropping* serta citra dengan nilai intensitas RGB.



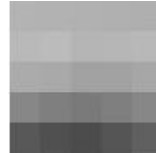
(a)



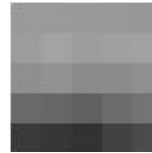
(b)



(c)



(d)



(e)

Gambar 5.2. Proses *Cropping* Citra. (a) Citra awal RGB, (b) Citra yang telah di *cropping*, (c) Citra *cropping Red channel*, (d) Citra *cropping Green channel*, (e) Citra *cropping Blue channel*

Diambil sebagian citra dengan proses *cropping* pada salah satu frame video 1 berupa matriks 5 x 5. Didapatkan matriks R,G,B dari citra tersebut seperti pada matriks dibawah ini.

$$r = \begin{bmatrix} 209 & 203 & 202 & 199 & 198 \\ 210 & 203 & 197 & 194 & 196 \\ 196 & 185 & 178 & 174 & 178 \\ 157 & 139 & 133 & 132 & 140 \\ 111 & 91 & 88 & 90 & 101 \end{bmatrix}, g = \begin{bmatrix} 179 & 178 & 177 & 176 & 175 \\ 185 & 187 & 181 & 182 & 184 \\ 171 & 196 & 162 & 162 & 167 \\ 135 & 127 & 122 & 127 & 135 \\ 88 & 80 & 76 & 85 & 96 \end{bmatrix}$$

$$, b = \begin{bmatrix} 154 & 154 & 153 & 152 & 151 \\ 157 & 163 & 157 & 160 & 162 \\ 143 & 146 & 139 & 140 & 144 \\ 106 & 103 & 97 & 108 & 116 \\ 59 & 55 & 51 & 66 & 76 \end{bmatrix}$$

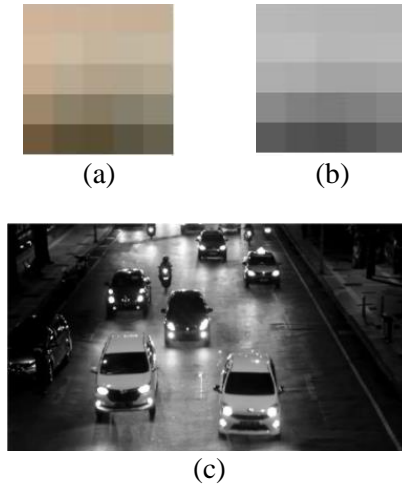
Proses *grayscale* dilakukan dengan menggunakan metode Luminosity yang mengalihkan setiap nilai RGB dengan konstanta yang telah ditentukan. Secara matematis dapat ditulis sebagai berikut.

$$grayscale = 0.21 * r + 0.72 * g + 0,07 * b$$

Citra yang akan dirubah menjadi citra *grayscale* didapatkan sebagai berikut

$$\begin{aligned}
 grayscale &= 0.21 \begin{bmatrix} 209 & 203 & 202 & 199 & 198 \\ 210 & 203 & 197 & 194 & 196 \\ 196 & 185 & 178 & 174 & 178 \\ 157 & 139 & 133 & 132 & 140 \\ 111 & 91 & 88 & 90 & 101 \end{bmatrix} + 0.72 \begin{bmatrix} 179 & 178 & 177 & 176 & 175 \\ 185 & 187 & 181 & 182 & 184 \\ 171 & 196 & 162 & 162 & 167 \\ 135 & 127 & 122 & 127 & 135 \\ 88 & 80 & 76 & 85 & 96 \end{bmatrix} \\
 &+ 0.07 \begin{bmatrix} 154 & 154 & 153 & 152 & 151 \\ 157 & 163 & 157 & 160 & 162 \\ 143 & 146 & 139 & 140 & 144 \\ 106 & 103 & 97 & 108 & 116 \\ 59 & 55 & 51 & 66 & 76 \end{bmatrix} \\
 &= \begin{bmatrix} 44 & 43 & 42 & 42 & 42 \\ 44 & 43 & 41 & 41 & 41 \\ 41 & 39 & 37 & 37 & 37 \\ 33 & 29 & 28 & 28 & 29 \\ 23 & 19 & 18 & 19 & 21 \end{bmatrix} + \begin{bmatrix} 129 & 128 & 127 & 127 & 126 \\ 133 & 135 & 130 & 131 & 132 \\ 123 & 122 & 117 & 117 & 120 \\ 97 & 91 & 88 & 91 & 97 \\ 63 & 58 & 55 & 61 & 69 \end{bmatrix} + \begin{bmatrix} 11 & 11 & 11 & 11 & 11 \\ 11 & 11 & 11 & 11 & 11 \\ 10 & 10 & 10 & 10 & 10 \\ 7 & 7 & 8 & 8 & 8 \\ 4 & 4 & 5 & 5 & 5 \end{bmatrix} \\
 &= \begin{bmatrix} 184 & 182 & 180 & 180 & 179 \\ 188 & 189 & 182 & 183 & 184 \\ 174 & 171 & 164 & 164 & 167 \\ 137 & 127 & 123 & 127 & 134 \\ 90 & 81 & 77 & 85 & 95 \end{bmatrix}
 \end{aligned}$$

Sehingga citra tersebut telah berubah menjadi citra *grayscale* yang dapat dilihat pada gambar dibawah ini

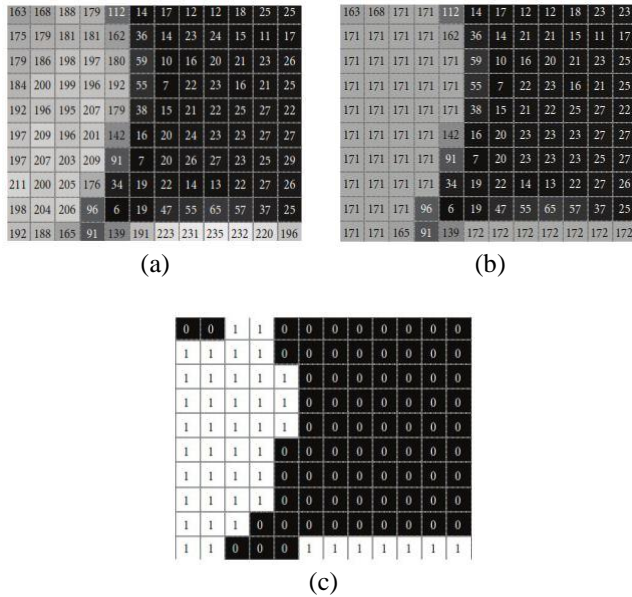


Gambar 5.3. Proses *Grayscale*. (a) Citra *cropping* RGB, (b), Citra *cropping grayscale*, (c) Citra *Grayscale*.

b. Proses Biner

Setelah didapatkan citra *grayscale*, kemudian masuk ke proses perubahan citra *grayscale* menjadi citra biner. Sehingga citranya menjadi citra dalam bentuk hitam putih.

Proses binerisasi yang dilakukan dengan menggunakan *Extended-Maxima transform* yang membuat *pixel-pixel* yang terdapat pada matriks citra yang memiliki nilai maximum akan nilainya menjadi 1 yang merupakan putih. Untuk yang nilainya rendah dirubah menjadi 0 yang merupakan warna hitam. Proses *Extended-Maxima transform* dapat dilihat seperti pada Gambar 5.4 dibawah ini.



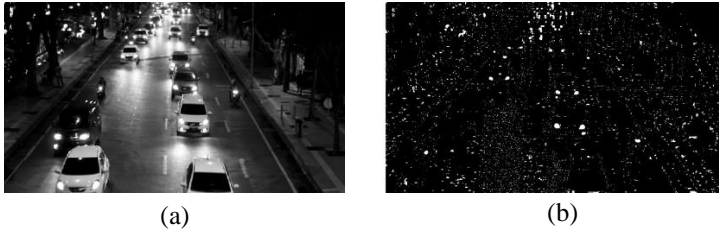
Gambar 5.4. Proses *Extended-Maxima Transform*. (a) citra grayscale, (b) *Maxima transform*, (c) *Extended-Maxima Transform*
(Sumber : “Extended-Maxima Transform Watershed Segmentation Algorithm for Touching Corn Kernels”[14])

Citra *grayscale* yang telah didapatkan sebelumnya dilakukan proses binerarisasi dengan menggunakan *Extended-Maxima transform* menjadi seperti pada Gambar 5.5 berikut ini.



Gambar 5.5. Proses *Extended-Maxima Transform*. (a) Citra *Grayscale cropping*, (b) Citra *Biner cropping*

Sehingga didapat citra biner seperti pada Gambar 5.6 dibawah ini.

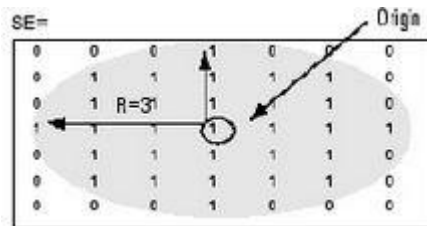


Gambar 5.6. Proses Biner (a) Citra *grayscale*, (b) Citra biner

c. Proses Operasi Morfologi & *Filtering Noise*

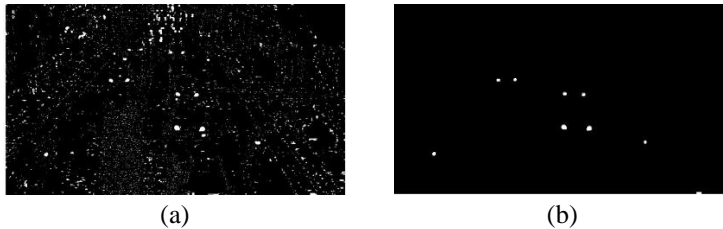
Pada citra biner yang telah didapatkan, dilakukan proses operasi morfologi yaitu proses opening yang juga berfungsi untuk penghilangan gangguan (*noise*) yang terdapat pada citra yang akan diteliti sehingga didapatkan hasil citra yang memiliki *noise* yang sedikit atau tidak ada sama sekali *noise*.

Proses operasi morfologi opening yang dilakukan dengan menggunakan *structuring element* berbentuk cakram (*disk*) dengan nilai value yang telah ditentukan.



Gambar 5.7. *Structuring Element* berbentu *disk*
(Sumber : Operasi Morfologi Citra Dengan Matlab[13].)

Sehingga citra biner yang telah di operasi morfologi dan penghilangan *noise* menjadi seperti pada Gambar 5.8.



Gambar 5.8. Proses *filtering noise*, (a) Citra biner, (b) Citra *Filtering*

d. Proses *Boundaries*

Untuk mendapatkan bagian supaya terlihat bahwa bagian tersebut adalah lampu dengan cara menentukan *Bouderies* yang berfungsi untuk mendapatkan tepian dari suatu objek.

Pada citra yang telah dilakukan operasi morfologi tsebelumnya didapatkan citra biner yang terdapat objek bulatan-bulatan yang merupakan bagian dari lampu kendaraan. Sehingga dengan melakukan proses *boundaries* didapatkan hasil seperti pada Gambar 5.9.

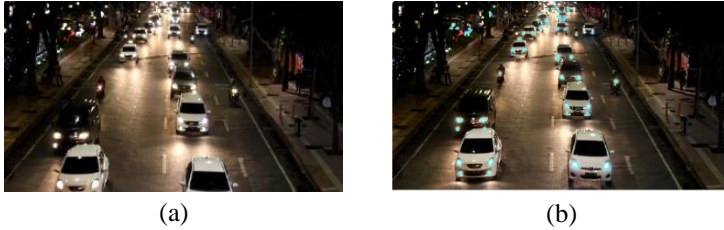


Gambar 5.9. Proses *Boundaries* (a) Citra proses *filtering noise*, (b) *Boundaries*

5.3.2. Proses Deteksi Bagian Lampu

Setelah proses *pre-processing* kemudian ditentukan bagian lampu dari kendaraan. Pada bagian yang telah didapatkan bagian tepi dari lingkaran lampu. Kemudian ditampilkan dengan memberikan warna pada bagian tepi lingkaran lalu ditampilkan

pada video yang akan ditampilkan. Sehingga akan menampilkan hasil seperti pada Gambar 5.10.



Gambar 5.10. Proses Deteksi Lampu. (a) Citra yang belum terdeteksi bagian lampu, (b) Citra yang sudah terdeteksi bagian lampu

5.3.3. Proses *Pairing*

Bagian lampu yang telah terdeteksi kemudian dilakukan proses *pairing*. Pada proses *pairing* akan dideteksi jarak (distance) terdekat pada setiap lampu (*blob*) kemudian ditampilkan garis penghubung pada setiap lampu yang memiliki jarak yang memenuhi.

Pada citra biner yang telah dilakukan proses *filtering noise* memiliki beberapa bulatan. Untuk mendapatkan titik untuk bisa dilakukannya proses *pairing* digunakan rumus *phytagoras* jarak antara 2 titik. Dengan menggunakan jarak antara 2 titik akan dicari jarak dari bulatan yang terdapat pada citra biner secara keseluruhan. Maka dari itu dibuat batas jarak yang akan digunakan untuk menghasilkan *pairing* yang baik. Jarak yang dihasilkan jarak yang telah ditentukan yaitu antara 50 sampai 125.

Sebagai contoh jika $x_1 = 471$, $x_2 = 560$ $y_1 = 175$ dan $y_1 = 26$. Maka untuk jarak yang didapat adalah :

$$\begin{aligned}
 distance &= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\
 &= \sqrt{(471 - 560)^2 + (175 - 26)^2} \\
 &= \sqrt{(-89)^2 + (149)^2}
 \end{aligned}$$

$$\begin{aligned}
&= \sqrt{7.921 + 22.201} \\
&= \sqrt{30.122} \\
&= 173,556907
\end{aligned}$$

Didapatkan jarak antar titik 173.556907. Maka pada jarak sebesar ini tidak dilakukan *pairing* dikarenakan tidak memenuhi batas jarak ditentukan.

Kemudian untuk $x_1 = 583$, $x_2 = 663$ $y_1 = 1$ dan $y_1 = 1$. Maka untuk jarak yang didapat adalah :

$$\begin{aligned}
distance &= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\
&= \sqrt{(583 - 663)^2 + (1 - 1)^2} \\
&= \sqrt{(-80)^2 + (0)^2} \\
&= \sqrt{6.400 + 0} \\
&= \sqrt{6.400} \\
&= 80
\end{aligned}$$

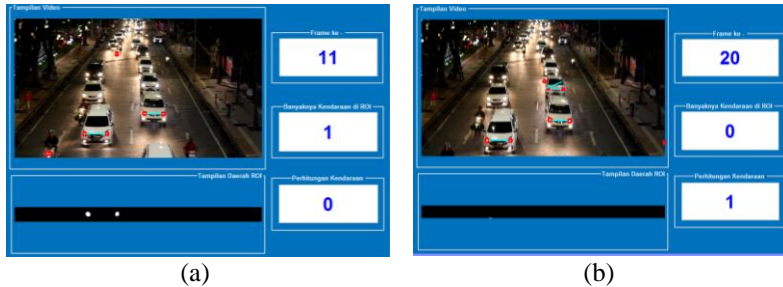
Didapatkan jarak antara titik 80. Maka pada jarak sebesar ini dilakukan *pairing* dikarenakan memenuhi batas jarak yang telah ditentukan. Pada Gambar 5.11 memperlihatkan hasil dari *pairing*



Gambar 5.11. Proses *pairing*. (a) Citra yang sudah terdeteksi bagian lampu, (b) Citra yang telah memiliki *pairing*

5.3.4. Proses *Counting*

Untuk proses perhitungan (*counting*) dilakukan dengan cara menghitung daerah lampu (*blob*) yang terdeteksi pada daerah ROI. Jika telah melewati daerah ROI maka perhitungan dilakukan. Untuk hasil perhitungan dihasilkan seperti pada Gambar 5.12.



Gambar 5.12. Proses *counting*. (a) Objek Lampu memasuki daerah ROI, (b) Objek Lampu telah melewati ROI

Pada Gambar 5.12 dapat terlihat bahwa kendaraan yang telah melewati daerah ROI, maka pada kotak perhitungan berubah dari 0 menjadi 1.

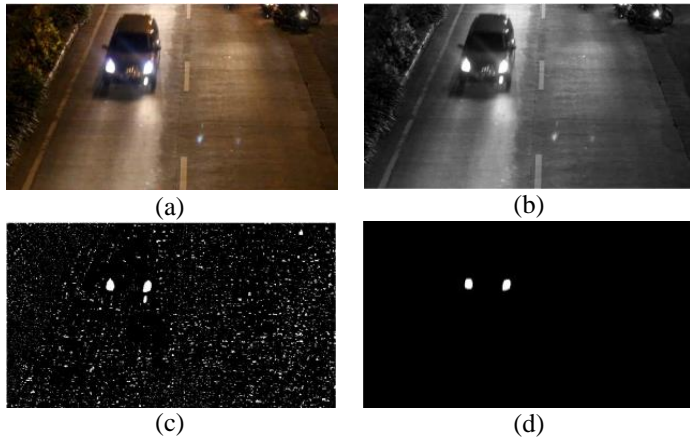
5.4. Uji Coba Video 1

Berikut adalah video kendaraan yang melewati jalan raya pada waktu malam hari. Video ini diambil di jembatan penyeberangan yang terdapat di Jalan Kenjeran, Surabaya pada sekitar pukul 7 malam. Kondisi jalan raya dengan pencahayaan yang terang.

5.4.1. Hasil proses *pre-processing*

Pada Gambar 5.13 memperlihatkan hasil dari salah satu frame pada Video 1 setelah dilakukan proses *pre-processing*. Terlihat bahwa hasil dari proses *pre-processing* pada frame

tersebut mendapatkan bulatan yang merupakan bagian lampu dari kendaraan tersebut.



Gambar 5.13. *Pre-processing* Video 1, (a) RGB, (b) *Grayscale*, (c) Biner, (d) *Filter noise*

5.4.2. Hasil Deteksi dan *Pairing*

Kemudian pada Gambar 5.14 memperlihatkan hasil deteksi dan pairing dari lampu kendaraan yang didapatkan sebelumnya pada proses *pre-processing*.



Gambar 5.14. Hasil Deteksi dan *Pairing* Video 1

5.4.3. Hasil Counting

Setelah didapatkan bagian lampu dan *pairing* dari lampu kendaraan. Kemudian dilakukan proses perhitungan kendaraan (*counting*) dari kendaraan yang melewati dan terdeteksi.



Gambar 5.15. Tampilan Akhir Hasil Video 1

Dari hasil akhir yang didapatkan pada proses Video 1 menghasilkan bahwa kendaraan yang terdeteksi 20 dari 12 kendaraan yang melewati selama durasi yang berlangsung. Itu dikarenakan bahwa cahaya yang terang dan membuat banyaknya *noise* yang terdapat pada video tersebut. Sehingga *noise* yang terdeteksi juga ikut terhitung pada proses perhitungan.

Pada tabel 5.2 memperlihatkan hasil perhitungan dan akurasi dari Video 1

Tabel 5.2. Akurasi Video 1
Jumlah Mobil

Sebenarnya	12
Terdeteksi	20
Akurasi	33.33 %

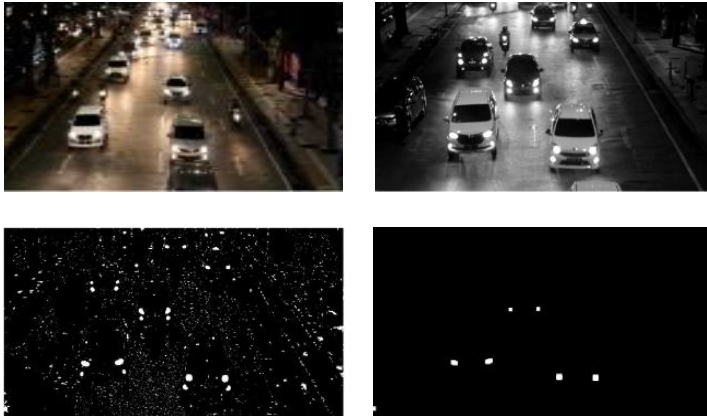
Sesuai dengan hasil yang didapatkan bahwa untuk tempat yang memiliki cahaya yang terang menghasilkan akurasi yang kecil dan *error* yang besar diakibatkan *noise* yang juga ikut terdeteksi.

5.5. Uji Coba Video 2

Berikut adalah video kendaraan yang melewati jalan raya pada waktu malam hari. Video ini diambil di jembatan penyeberangan yang terdapat di Surabaya Plaza, Embong Kaliasin, Genteng Surabaya pada sekitar pukul 8 malam. Kondisi jalan raya dengan pencahayaan yang cukup.

5.5.1. Proses *pre-processing*

Pada Gambar 5.16 memperlihatkan hasil dari salah satu frame pada Video 2 setelah dilakukan proses *pre-processing*. Terlihat bahwa hasil dari proses *pre-processing* pada frame tersebut mendapatkan bulatan yang merupakan bagian lampu dari kendaraan tersebut.



Gambar 5.16. Proses *pre-processing* Video 2, (a) RGB, (b) *Grayscale*, (c) Biner, (d) *Filter noise*

5.5.2. Hasil Deteksi dan *Pairing*

Kemudian pada Gambar 5.17 memperlihatkan hasil deteksi dan pairing dari lampu kendaraan yang didapatkan sebelumnya pada proses *pre-processing*.



Gambar 5.17. Hasil Deteksi dan *Pairing* Video 2

5.5.3. Hasil *Counting*

Setelah didapatkan bagian lampu dan *pairing* dari lampu kendaraan. Kemudian dilakukan proses perhitungan kendaraan (*counting*) dari kendaraan yang melewati dan terdeteksi.



Gambar 5.18. Tampilan Akhir Hasil Video 2

Dari hasil akhir yang didapatkan pada proses Video 2 menghasilkan bahwa kendaraan yang terdeteksi 26 dari 25 kendaraan yang melewati selama durasi yang berlangsung. Dengan pencahayaan yang cukup pada jalan di video 2 membuat *noise* pada Video 2 sedikit. Sehingga *noise* yang terdeteksi juga sedikit terhitung pada proses perhitungan.

Pada tabel 5.3 memperlihatkan hasil perhitungan dan akurasi dari Video 2

Tabel 5.3. Akurasi Video 2
Jumlah Mobil

Sebenarnya	25
Terdeteksi	26
Akurasi	96 %

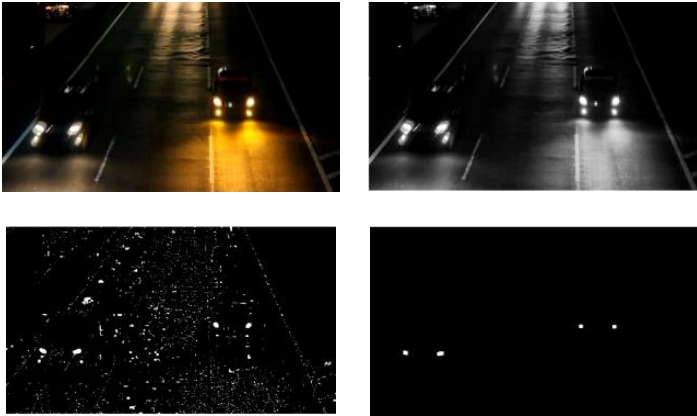
Sesuai dengan hasil yang didapatkan bahwa untuk tempat yang memiliki cahaya yang cukup menghasilkan akurasi yang besar dan *error* yang kecil menghasilkan akurasi yang baik.

5.6. Uji Coba Video 3

Berikut adalah video kendaraan yang melewati jalan raya pada waktu malam hari. Video ini diambil di jembatan jalan tol Surabaya-Porong yang terdapat di Jalan Manunggal Kebonsari, Kebonsari, Jambangan, Surabaya pada sekitar pukul 9 malam. Kondisi jalan raya dengan pencahayaan yang kurang.

5.7.1. Proses *pre-processing*

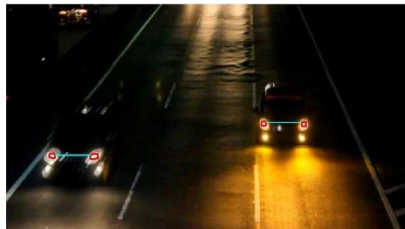
Pada Gambar 5.19 memperlihatkan hasil dari salah satu frame pada Video 2 setelah dilakukan proses *pre-processing*. Terlihat bahwa hasil dari proses *pre-processing* pada frame tersebut mendapatkan bulatan yang merupakan bagian lampu dari kendaraan tersebut.



Gambar 5.19. *pre-processing* Video 3, (a) RGB, (b) *Grayscale*, (c) Biner, (d) *Filter noise*

5.7.2. Hasil Deteksi dan Pairing

Kemudian pada Gambar 5.20 memperlihatkan hasil deteksi dan pairing dari lampu kendaraan yang didapatkan sebelumnya pada proses *pre-processing*.



Gambar 5.20. Hasil Video 3

5.5.4. Hasil *Counting*

Setelah didapatkan bagian lampu dan *pairing* dari lampu kendaraan. Kemudian dilakukan proses perhitungan kendaraan (*counting*) dari kendaraan yang melewati dan terdeteksi.



Gambar 5.21. Tampilan Akhir Hasil Video 3

Dari hasil akhir yang didapatkan pada proses Video 3 menghasilkan bahwa kendaraan yang terdeteksi 14 dari 19 kendaraan yang melewati selama durasi yang berlangsung. Dengan pencahayaan yang kurang pada jalan di video 3 membuat *noise* pada Video 3 sedikit. Selain itu pada video 3, kendaraan yang melewati jalan berjalan dengan kecepatan yang cepat. Sehingga *noise* yang terdeteksi sedikit dan ada kendaraan yang tidak terdeteksi.

Tabel 5.4. Akurasi Video 3
Jumlah Mobil

Sebenarnya	19
Terdeteksi	14
Akurasi	73,68 %

Sesuai dengan hasil yang didapatkan pada Tabel 5.4 bahwa untuk tempat yang memiliki cahaya yang kurang menghasilkan akurasi yang cukup besar dan *error* yang kecil menghasilkan akurasi yang baik.

BAB VI PENUTUP

6.1. Kesimpulan

Berdasarkan analisis terhadap hasil pengujian program, maka dapat diambil kesimpulan sebagai berikut :

1. Program dapat mendeteksi kendaraan yang terdapat pada video dengan menggunakan Pengolahan Citra Digital berdasarkan lampu kendaraan dengan tahapan pre-processing (grayscale, biner dan operasi morfologi), *pairing* dan *counting* yang digunakan.
2. Proses *pairing* yang digunakan untuk menghubungkan lampu yang telah dilakukan proses pre-processing dapat digunakan sehingga lampu kendaraan dapat terhubung seperti pada hasil tugas akhir ini.
3. Perhitungan jumlah kendaraan memberikan hasil yang cukup baik yang dapat dilihat berdasarkan objek lampu yang terdeteksi. Hasil perhitungannya yaitu pada Video 1 dengan kondisi cahaya yang terang menghasilkan akurasi 33,33%. Video 2 dengan kondisi cahaya yang cukup menghasilkan akurasi 96%. Video 3 dengan kondisi cahaya yang kurang menghasilkan akurasi sebesar 73,68%

6.2. Saran

Ada beberapa hal yang penulis sarankan untuk pengembangan penelitian selanjutnya :

1. Program ini hanya bisa mendeteksi objek kendaraan, tidak dapat untuk mengklasifikasikan jenis dari kendaraan. Maka untuk kedepannya bisa dilakukan klasifikasi untuk membedakan kendaraan yang lewat di jalan raya pada waktu malam hari.
2. Program ini hanya melakukan pendeteksian terhadap mobil saja. Untuk kedepannya dapat mendeteksi kendaraan selain mobil.

3. Program ini masih memiliki eror berupa masih mendeteksi noise yang ada dikarenakan kondisi yang berbeda. Untuk kedepannya bisa menggunakan metode lain untuk memperkecil noise yang ada sehingga mendapatkan error yang kecil.

DAFTAR PUSTAKA

- [1] Anonim. 2016. “Jumlah Kendaraan Bermotor tahun 1949 - 2014”. <http://data.go.id/dataset/jumlah-kendaraan-bermotor-unit/resource/f9c24882-8de4-481e-9cb6-400ed8fbb0df>, *posted* : 2016. Diakses pada tanggal 07-02-2017.
- [2] Robert, K. 2009. “Video-based traffic monitoring at day and night time vehicle detection and tracking”. IEEE International Conference on Intelligent Transportation Systems, pp. 1-6.
- [3] Anonim. 2013. “Kisah Edi Warsito: Hitung Kendaraan Pemudik Secara Manual”. <http://news.liputan6.com/read/662163/kisah-ediwarsito-hitung-kendaraan-pemudik-secara-manual>, *posted* : 10/08/2013 14:43. Diakses pada tanggal 8-02-2017.
- [4] Anonim. 2013. “Ada Alat Canggih untuk Hitung Kendaraan di Posko Nagreg”. <http://ramadan.okezone.com/read/2013/08/05/335/847739/a-da-alat-canggih-untuk-hitung-kendaraan-di-posko-nagreg>, *posted* : 05/08/2013 15:40. Diakses pada tanggal 8-02-2017.
- [5] Robert, K. 2009. “Night-Time Traffic Surveillance A Robust Framework for Multi-Vehicle Detection, Classification and Tracking”. International Journal of Computer Technologi and Application, Volume 5 Issues 2.
- [6] Padmavathi, S. dan Gunasekaran, K. 2014. “Night Time Vehicle Detection for Real Time Traffic Monitoring Systems: A Review”. International Journal Computer Technologi and Applications, Volume 5(2),451-456.
- [7] Hariyanto, Z. 2014. “Klasifikasi Jenis Kendaraan Bergerak Berbasis *Geometric Invariant Moment*”. Tugas Akhir. Jurusan Matematika ITS.

- [8] Padmavathi,S., Naveen, C. R. dan Kumari, V., H. 2016. "Vision based Vehicle Counting for Traffic Congestion Analysis during Night Time". Indian Journal of Science and Technology, Vol 9(20).
- [9] Gonzalez, R.C., dan Woods, R.E. 2002. "Digital Image Processing". United States of America: Tom Robbins Publisher.
- [10] Al Bovik. 2000. "Handbook of Image and Video Processing". San Diego: Academic Press Publisher.
- [11] Anonim. 2015. "Defenisi Pengertian Citra". <http://www.definisi-pengertian.com/2015/07/definisi-pengertian-citra-image.html>, *posted* : 07/2015. Diakses pada tanggal 9-02-2017.
- [12] Fajriyah F. 2016. "Pengembangan Perangkat Lunak Deteksi Kecepatan Kendaraan Bergerak Berbasis Pengolahan Citra Digital". Tugas Akhir. Jurusan Matematika ITS.
- [13] Rifki Nugroho, Muhammad. 2011. "Operasi Morfologi Citra Dengan Matlab". Departemen Teknik Elektro, Fakultas Teknik Universitas Indonesia. Depok.
- [14] Qin, Y. Wang, W. Liu, W. dan Yuan, N. 2013. "Extended-Maxima Transform Watershed Segmentation Algorithm for Touching Corn Kernels". Hindawi Publishing Corporation Advances in Mechanical Engineering.

BIODATA PENULIS



Penulis bernama lengkap **Gery Dias Claudio**, lahir di Bukittinggi, 13 November 1994. Anak kelima dari pasangan Afliyus dan Renny Idrus, serta memiliki kakak perempuan Febrina Yolanda, Ranny Yolanda, Mealya Fridayana dan Pretty Sakta Riny. Penulis mengikuti pendidikan dasar dari Sekolah Dasar hingga Sekolah Menengah Atas di Kota Bukittinggi. Penulis menempuh pendidikan di SD Negeri 02 Campago Guguak Bulek, SMP Negeri 5 Bukittinggi, dan SMA Negeri 4 Bukittinggi. Setelah Lulus dari SMAN 4 Bukittinggi pada tahun 2013 yang lalu, penulis melanjutkan pendidikan tingginya di Institut Teknologi Sepuluh Nopember (ITS) Surabaya dengan mengambil Jurusan Matematika dengan bidang minat Ilmu Komputer. Selama mengikuti perkuliahan di ITS, penulis turut aktif dalam beberapa kegiatan kemahasiswaan sebagai, staff Departemen Sains, Teknologi, dan Keprofesian Himatika ITS Periode 2014/2015, dan Team Konseptor OMITS 2016 Department Himatika ITS Periode 2015/2016. Aktif pada Lembaga Dakwah Jurusan Ibnu Muqhlah pada Departemen Dana Usaha selama dua periode yaitu periode 2014/2015 dan periode 2015/2016. Selain aktif dalam beberapa kegiatan kemahasiswaan, penulis juga mengikuti Kerja Praktek PT.INTENS (Inti Konten Indonesia) yang bertempat di kota Bandung. Informasi lebih lanjut mengenai Tugas Akhir ini dapat ditujukan ke penulis melalui email : gerydiasclaudio@gmail.com.